PhD thesis
Wavelets and ill posed problems: optic flow and scattered data interpolation

Christophe BERNARD

Final version

REMERCIEMENTS

Je tiens tout d'abord à exprimer ma profonde reconnaissance pour mon directeur de thèse Stéphane Mallat, qui a consacré à l'encadrement de ma thèse un temps et une disponibilité d'esprit considérables, auxquels j'ai été d'autant plus sensible que son emploi du temps est très chargé. J'ai ainsi largement pu profiter de sa grande acuïté scientifique et de son enthousiasme indéfectible et communicatif pour le travail de ses étudiants. Je lui suis donc redevable d'avoir pu faire une thèse dans des conditions exceptionnelles.

Je remercie également Jean-Jacques Slotine, qui m'a accompagné tout au long de ce travail de thèse, au cours de longues et fructueuses discussions à Paris ou à Boston, où il m'a reçu de nombreuses fois.

Je tiens à remercier Albert Benveniste, Patrick Bouthémy et Ronald DeVore d'avoir accepté la tâche ingrate de lire ma thèse et d'écrire un rapport. Albert Cohen a très tôt manifesté un grand intérêt pour l'ensemble de ma thèse, et a pris la peine de la lire en détail et de me faire part de ses commentaires. Je lui en suis très reconnaissant. Jean Serra a également montré son intérêt pour les divers volets de mon travail et m'a donné l'occasion de le présenter au centre de morphologie mathématique, dont il est le fondateur.

Je remercie enfin Patrick Bouthémy, Ronald DeVore et Ingrid Daubechies de m'avoir donné l'occasion de présenter mon travail à différents séminaires qu'ils organisent.

J'ai eu la chance de faire ma thèse au CMAP, un centre de recherches accueillant et vivant, où règne une ambiance conviviale et dynamique. Parmi les artisans de ce bel équilibre, je dois d'abord citer Geo Boléat, d'un abord bourru, mais qui a un cœur d'or, ainsi que Jeanne Bailleul, toujours gaie, active et ardente défenseuse de la veuve et de l'orphelin. Je dois également parler de Liliane Doaré, pilier du CMAP et de Nathalie Limonta, une nouvelle recrue.

Je remercie Jean-Claude Nédélec, directeur au long cours du centre, qui gère ses troupes avec la bienveillance d'un père de famille. Pierre-Arnaud Raviart qui a régné sur le CMAP avec une voix de stentor, Vincent Giovangigli.

Je salue Carl Graham, grand spécialiste de la bible, qui élève heureusement le niveau de la discussion que François Jouve amène périodiquement au-dessous de la ceinture, Eric Bonnetier, qui se pique d'aimer le bon vin et les bons mots, Marc Schoenauer, Pedro Ferreira, Josselin Garnier, Aldjia Mazari, qui m'a spontanément proposé de faire une répétition de soutenance qui m'a été extrêmement utile. Je salue également Habib Ammari, Kamel Hamdache, Toufic Abboud, Frédéric Nataf, Robert "Bob" Brizzi et Geneviève Allain. Je remercie Jean-François Colonna qui m'a consacré du temps et de la patience pour réaliser un film de présentation, malgré un emploi du temps chargé.

Je salue les autres membres de l'équipe ondelettes, Emmanuel Bacry, Maureen Clerc, Jérôme Kalifa qui vient de convoler en justes noces, Rémi Gribonval, Erwan Le Pennec, Jérôme Fraleu, Yann Samuelides, ainsi que tous les autres membres du CMAP qui ont ajouté leur pierre à l'harmonie de l'équipe, à savoir Shiraz Latiri qui nous a souvent conduits, Rita et moi, dans sa shirazmobile, Sofiane Oussedik, Daniel Delahaye, Annalisa Ambroso, Paul Seignourel, Florence Bailly, Natacha Vialle, Rama Cont et ses plaisanteries homéopathiques, Erik Burmån et Jonås Ribbe, Denis Barbier et Snorre Christiansen sans oublier mes collègues de bureau Alain Racine, Sana Ben Hamida et Jean-Philippe Ayanidès.

Je dédie cette thèse à Rita, et à mes parents.

Contents

Tout événement est écrit dans les astres et les phénomènes cycliques de la nature. Il faut chercher ceux de ces phénomènes qui y participent, et dans quelle mesure chacun influe sur l'événement. Cette tâche est difficile, mais possible car le destin de chaque chose est étroitement liée à une conjugaison de ces cycles.

Fu-Hi,
Traité d'astrologie apocryphe,
28^{e}siècle av JC.

Il résulte [...] que si l'on propose une fonction $f\,x$, dont la valeur est représentée dans un intervalle déterminé, depuis $x = 0$ jusqu'à $x = X$, par l'ordonnée d'une ligne courbe tracée arbitrairement on pourra toujours développer cette fonction en une série qui ne contiendra que les sinus, ou les cosinus, ou les sinus et les cosinus des arcs multiples ou les seuls cosinus des multiples impairs.

Joseph Fourier,
Théorie analytique de la chaleur,
Chap III, Section VI, Art. 235 (1822).

Chapter 1

Introduction

This PhD thesis handles two separate problems that have two things in common. They are both ill-posed problems, and for each of these problems, we suggest a solution that is based on wavelets. The first problem is that of optic flow estimation. The second problem is that of scattered data interpolation. This thesis can therefore be seen as two illustrations of ill posed problem solving with wavelets.

What is an inverse or ill-posed problem? It is a problem that consists in solving an equation

$$Ax = y$$

where A is a non invertible operator. We consider in all what follows that A is a linear operator mapping a vector space E to a vector space F.

A general way to handle ill-posed problems is regularization. Regularization consists in rewriting the inversion problem as a minimization of a quadratic matching functional

$$x = \arg\min_{x'} \|Ax' - y\|^2$$

that is also ill-posed, because this functional is positive, but not positive definite. The regularization method then consists in assuming that the solution is in a Hilbert space $\mathcal{H} \subset E$ and in penalizing the matching functional with an additional term that is strictly positive definite.

$$x = \arg\min_{x'} \|Ax' - y\|^2 + \lambda \|x'\|^2_{\mathcal{H}}$$

The resulting functional is then positive definite by construction, and the minimization problem thus has a unique solution, provided that the operator A is continuous w.r.t the norm $\|.\|_{\mathcal{H}}$.

Regularization is thus a systematic way to transform an ill-posed problem into a well posed one. This advantage has a high price. Regularizing consists in doing a strong assumption on the original signal x, that:

- it is in the Hilbert space $\mathcal{H}$

- the solution to the problem is a Bayesian optimum based on the assumption that x is a realization of a Gaussian random process of covariance given by the scalar product in $\mathcal{H}$.

These assumptions may be sometimes difficult to motivate. Moreover, the choice a given Hilbert space is sometimes arbitrary, while it has a strong influence on the shape of the solution. In each problem we study in this thesis, our approach also consists in doing some local regularization. The difference is that the regularity hypothesis is not fixed in advance, but varies locally according to the signal itself.

1.1 Optic flow measurement

The problem we handle in the first part of this thesis is that of optic flow measurement in a video sequence. In general, the evolution of the image in time is caused by two main factors

- sudden changes between two successive sequences, that are usually relatively rare,
- the relative motion between objects in the scene and the camera.

The relative motion between objects and the camera is a three dimensional vector field in the reference frame of the camera we call the real motion. The scene is then projected to the camera image plane, and we can thus define a second vector field: the projected motion field. Let p be the projection operator of the camera (either linear or projective). For each point $\boldsymbol{x}$ of the image, that is the projection $p(\boldsymbol{X})$ of a real point $\boldsymbol{X}$ of real velocity $\boldsymbol{V}$, the optic flow is the projected motion $\boldsymbol{v} = dp(\boldsymbol{X})\boldsymbol{V}$.

The purpose of optic flow measurement is only to estimate this motion in the image plane from the knowledge of the images sequence $I(t;\boldsymbol{x})$. This problem is in itself already ill-posed. A natural continuation of this problem is that of recovering the real 3–dimensional motion, which we will not tackle here.

The optic flow measurement has several applications. It can be used as such as an image deformation model to store efficiently image sequences by motion compensation. Optic flow measurement can also be used for 3-dimensional scene analysis, if we use additional models (of rigidity or of solid motion). This can be used for 3 dimensional model reconstruction for virtual reality, or in robotics to built a representation of the environment of a moving robot.

A sequence of black and white images is represented by a grey level function $I(t;x_1,x_2)$. To measure the underlying optic flow, it is usual to make an assumption of constant illumination. This assumption consists in saying that a real point projected on the camera image plane has a grey level that does not change in time.

The time derivative of this grey level is

$$\frac{d}{dt}I(t;\boldsymbol{x}(t)) = \frac{\partial I}{\partial t} + \boldsymbol{v}\cdot\boldsymbol{\nabla} I$$

and the constant illumination assumption thus provides the following equation

$$\frac{\partial I}{\partial t} + \boldsymbol{v}\cdot\boldsymbol{\nabla} I = 0 \qquad \text{(OF)}$$

we call "optic flow equation".

This motion is not always measurable, nor even visible by the human eye. If we consider for example a uniform area of the image that is thus translation invariant, different motions are translated into the same local image changes. The problem then does not have a unique solution. This is called the *aperture* problem.

1.1.1 Motion estimation as an ill-posed problem

If we consider the above optic flow equation, we can see that the value of the vector $\boldsymbol{v}(\boldsymbol{x}_0, t_0)$, (of dimension usual equal to 2) is constrained by a single scalar equation. In such a case, the only information we can extract on the field $\boldsymbol{v}$ is its component parallel to the image gradient, while the orthogonal component is unknown.

Since this problem is as such not solvable, several approaches have been suggested that all rely on an additional assumption on the flow.

Regularization

A regularization method has been suggested by Horn and Schunck in 1980. It consists in looking for a solution for $\boldsymbol{v}$ at a given time as a minimum of the following functional

$$\boldsymbol{v} = \arg\,\min \left(\iint \left(\frac{\partial I}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} I \right)^2 dx_1 dx_2 + \lambda \|\boldsymbol{v}\|_{\mathcal{H}}^2 \right)$$

where $\|.\|_{\mathcal{H}}$ is a regularizing Hilbert norm (like a Sobolev norm). This consists in assuming that the best solution to this problem is the smoothest one. In practice, the numerical resolution of this problem is done by inverting a symmetric linear system of very large size (with $2NM$ unknowns if the image size is $N \times M$) which is inverted by iterations.

Matching methods

Another approach (block matching method) consists in trying to match windows of two successive frames. If $W + \{\boldsymbol{x}_0\}$ is a space window around the points $\boldsymbol{x}_0$, the matching method consists in looking for a vector $\boldsymbol{v}(\boldsymbol{x}_0, t)$ such that the quadratic error

$$\iint_{W+\{\boldsymbol{x}_0\}} \left(I(t+1; \boldsymbol{x} + \boldsymbol{v}) - I(t; \boldsymbol{x}) \right)^2 d^2\boldsymbol{x}$$

is minimum. We have to assume for this that the optic flow is constant over the window $W + \{\boldsymbol{x}_0\}$. It is however costly because the functional to be minimized is non convex and is likely to have many local minima. We thus have to scan a whole range of discrete displacements $\boldsymbol{v} \in \mathcal{G}$.

Spatiotemporal filtering methods

Another family of techniques have been derived on the basis of observations made by Adelson and Bergen. They consist in doing a local spectral analysis of time-space windows of the

sequence. If over such a small window, the displacement is uniform:

$$I(t; x_1, x_2) = P(x_1 - v_1 t, x_2 - v_2 t)$$

then the Fourier transform of this function window is a Dirac ridge whose support is orthogonal to the velocity vector:

$$\hat{I}(\tau; \xi_1, \xi_2) \propto \hat{P}(\xi_1, \xi_2)\delta(v_1\xi_1 + v_2\xi_2 + \tau)$$

By identifying the tilt of the plane of equation $v_1\xi_1 + v_2\xi_2 + \tau = 0$, we can recover the components of the velocity (v_1, v_2). Variants of these technique exist, depending on whether the modulus or the phase output of the time-frequency filters is used. These methods rely on an assumption that the optic flow is constant over parallelepipedic spatiotemporal windows of the picture sequence.

Filtered differential approach.

Weber and Malik have suggested to apply the optic flow equation to several filtered versions $I * f_n$ of the image sequence instead of the single original one I. If we can assume that the flow is constant over the support of all filters f_n, we can then write

$$\begin{aligned} I_n(\boldsymbol{x}, t) &= I * f_n(\boldsymbol{x}, t) \\ \frac{\partial I_n}{\partial t}(\boldsymbol{x}, t) + \boldsymbol{v}(\boldsymbol{x}, t) \cdot \boldsymbol{\nabla} I_n(\boldsymbol{x}, t) &= 0 \qquad\qquad n = 1 \dots N \end{aligned}$$

and thus obtain a set of N equations to solve for $\boldsymbol{v}$ with a least square method.

1.1.2 Time aliasing versus measure localization

In practice, the video sequences are sampled in all variables x_1, x_2 and t. Since the sequence is sampled in time, the picture values are only known for discrete values of t, so we do not measure velocities, but finite displacements. Most of the above described methods are local. To estimate the displacement at a given location $\boldsymbol{x}_0$, they only use grey values of the picture within a spatial window $W + \{\boldsymbol{x}_0\}$, where W is

- the support of the local Fourier analysis in spatiotemporal filtering methods,
- the support of the filters for the method of Weber and Malik,
- the support of the spatial derivation filters for the method described by Horn and Schunck (which is also subject to time aliasing although its formulation is global).

If the displacement $|\boldsymbol{d}|$ between two successive frames is larger than the window W, then the considered picture windows are

$$\begin{aligned} [I(t; \boldsymbol{x})]_{\boldsymbol{x} \in W} & \qquad \text{at time } t \\ [I(t+1; \boldsymbol{x})]_{\boldsymbol{x} \in W} = [I(t; \boldsymbol{x} - \boldsymbol{d})]_{\boldsymbol{x} \in W} & \qquad \text{at time } t+1 \end{aligned}$$

that correspond to disjoint portions of the original image $I(t)$, and thus do not allow to estimate the underlying motion. In block matching methods, where the window is moving, this appears also, but in terms of increased computational complexity. The set of possible displacements $\boldsymbol{v}$ in which we look for a best estimation has to be bounded and very large displacement are thus expensive to estimate.

On the other hand, we cannot work with arbitrarily large windows, because the optic flow estimations we get are then very correlated in space. We thus have to find a trade-off between two antagonistic constraints:

- the aliasing constraint, that requires that the measure is done on windows that are larger than the real displacement between two successive frames
- the need for a displacement map of the finest resolution possible, that requires to use small windows.

Multiresolution approaches

These considerations have led to the use of multiresolution variants of the classical techniques. These variants usually consist in performing measurements with windows of several size, and to combine the resulting measures by arbitrating between them or by refinement.

The method described by Weber and Malik is a multiresolution differential approach. The filters f_n are dilated on a range of geometric scales (powers of 1.8). They obtain displacement maps at several resolutions and select at each location the finest measure that is not impaired by time aliasing.

Etienne Mémin and Patrick Pérez describe a multiresolution variant of Horn and Schunck's method where the matching functional and the smoothness functional are not convex, in order to preserve optic flow discontinuity. The functional minimization is done at successive scales with several iterations at each scale. Their method shows to be fairly accurate (error about 20% less than the method presented in this thesis), but its computational burden is significantly higher.

Block matching methods also have multiresolution counterparts. The matching is done on filtered images (with a Laplacian pyramid, for example). Coarse scale displacements are reused at finer scales to choose a range $\mathcal{G}$ of displacements in which to look for a best motion estimation that is shifted around the coarse scale estimate. The gain of the multiresolution approach is to reduce the computational cost for a give range of motions, or to enlarge the range for a given computational cost.

Our approach: projected differential method

The approach we suggest and describe in this thesis can be compared with that of Weber and Malik. It consists in choosing as filters f_n wavelets. We start from a set of N mother wavelets $(\psi^n)_{n=1\ldots N}$, and build a wavelet basis or frame indexed by $n \in \{1, \ldots, N\}$, $j \in \mathbb{Z}$ and $\boldsymbol{k} \in \mathbb{Z}^2$ with

$$\psi^n_{j\boldsymbol{k}} = 2^j \psi^n(2^j \boldsymbol{x} - \boldsymbol{k})$$

The same way as Weber and Malik, we project the optic flow equation on these wavelets

$$\frac{\partial \langle I, \psi^n_{j\boldsymbol{k}} \rangle}{\partial t} + \left\langle \boldsymbol{v} \cdot \boldsymbol{\nabla} I, \psi^n_{j\boldsymbol{k}} \right\rangle = 0$$

If we now assume that the displacement $\boldsymbol{v}$ is uniform on the support of the wavelets $\psi^n_{j\boldsymbol{k}}$ for $n = 1 \dots N$ (scale separation hypothesis), these equations can be written

$$\frac{\partial \langle I, \psi^n_{j\boldsymbol{k}} \rangle}{\partial t} + \boldsymbol{v} \cdot \left\langle \boldsymbol{\nabla} I, \psi^n_{j\boldsymbol{k}} \right\rangle = 0$$

and after an integration by parts

$$\frac{\partial \langle I, \psi^n_{j\boldsymbol{k}} \rangle}{\partial t} = v_1 \left\langle I, \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_1} \right\rangle + v_2 \left\langle I, \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_2} \right\rangle$$

For given indices j and $\boldsymbol{k}$, we thus have a system of N equations whose unknowns are v_1 and v_2. The system is now overdetermined and we can for example extract a unique regression solution. v_1 and v_2 are the coordinates of the optic flow that is supposed to be uniform over a domain centered around $2^{-j}\boldsymbol{k}$ and of radius 2^{-j}. So far, this approach is very close to that of Weber and Malik.

It has however a number of advantages:

1. the computation algorithm is extremely fast. The computation time to estimate the optic flow map between two successive pictures is of the order of the number of pixels. Indeed, the different small systems we build are less numerous at coarse scales than at fine scales. We thus take advantage of the high correlation of coarse scale estimations by subsampling them. Moreover, all the coefficients of these systems are wavelet coefficients of the picture or of its time derivative. Their coefficients can thus be computed in a time proportional to the number of pixels in a frame.

2. the algorithm convergence can be proved. Under some assumptions (smoothness of the flow and richness of the underlying moving texture), we can show that this flow estimation technique is asymptotically exact.

3. we last show how this approach can be adapted to a number of different local optic flow models. We detail how to measure local illumination changes and thus obtain an optic flow estimation that is very robust w.r.t such changes.

1.1.3 Road map for the first part

In Chap. 2, we introduce some basic notations on the wavelets we use in this work. In chapter 3, we detail some characteristics of the optic flow measurement problem. We then describe our approach by comparing it to existing methods. We notably explain in detail the different ways to combine measures obtained at different scales. Whatever the chosen approach, the complexity order of the whole estimation process is always proportional to the number of pixels in a frame. It is thus much faster than the approach described by Weber and Malik. This method has some similarities with techniques developed by Simoncelli and

by Magarey and Kingsbury. We detail in Chap. 3 how our method differs from these two methods.

In chapter 5, we prove that our estimation technique is convergent. More precisely,

- if the condition number of the system matrices stays within a reasonable range of values (which means that the underlying picture contains enough details to extract its motion)
- if the motion field is smooth enough
- if the picture at a given point is of fixed (not necessarily known) Lipschitz smoothness index

then the estimation error goes to 0 as

- the measuring scale goes to 0
- the time interval between two successive images goes to 0 and is asymptotically negligible w.r.t the measuring scale (is bounded by some multiple of $2^{-(1+\theta)j}$).

This result is important because it shows that in some cases, we can estimate accurately the displacement that caused the changes in the picture sequence, although this was supposed to be an ill-posed problem. "In some cases" means that the underlying textures have to be rich enough in order for their motion to be visible. Such a requirement is truly necessary, since the motion of a uniform pattern can clearly not be detected.

Chap. 6 focuses on the numerical validation of our approach, and shows that is has results very comparable to competing methods, measures illumination changes and is more robust w.r.t these changes. This chapter also briefly describes the application to video compression (that has been implemented in two student projects), and mentions possible extensions and prospects for this method.

1.2 Multivariate scattered data interpolation

In a second part, we study the problem of building an interpolating function from scattered samples of an unknown function, which is sometimes called *learning problem*. We try to find a mapping from E to F that is fitted to some measures $(x_n, y_n)_{n=1\dots N}$ where $x_n \in E$ and $y_n \in F$:

$$f(x_n) = y_n \qquad \text{for all } n = 1 \dots N$$

When the function space $\mathcal{H} \subset F^E$ in which the unknown function is supposed to live is of infinite dimension, this problem is ill-posed, because a finite number of constraints cannot determine an infinite number of parameters.

Such problems of finding a good approximation of a function on the basis of some samples has various applications. To name a few, we can cite the simulation of complex dynamical systems, the reconstruction of surfaces for image synthesis and particularly in medical imaging, the resolution of partial differential equations etc.

The characteristics of this problem are very different, depending on the dimension of the input space S in which the measuring points x_n are. When this dimension is very small (1 to 5) we can call this function approximation or interpolation. If however the space dimension is very high, these problems are usually called learning problems, and the most popular ways to solves them are then radial basis function networks or neural networks. We give a brief description of these methods in Chap. 7.

Regularization

A regularization approach consists in choosing as a solution to this problem the function f in a Hilbert space $\mathcal{H}$ that fulfills the interpolation constraints and is of minimum norm $\|f\|_{\mathcal{H}}$. The solution can then be written as a linear combination of kernels

$$f(x) = \sum_{n=1}^{N} c_n K(x_n, x)$$

When the norm $\|.\|_{\mathcal{H}}$ is translation invariant and isotropic, the kernels $K(x_n, x)$ can be written as

$$K(x_n, x) = g(\|x_n - x\|)$$

and we have a radial basis function interpolation.

This method has a number of advantages

- the regularized problem is well posed and the solution is a continuous function of the sample output values y_n;
- the complexity of the solution f is not higher that the set of measures;
- this computational framework can be used whatever the starting space E, thanks to theorems of Schoenberg and Micchelli.
- this model has a Bayesian justification. The solution f is maximum of an a posteriori probability density if we make the two following assumptions: f is a realization of a centered Gaussian process of covariance $E(f(x)f(x')) = K(x, x')$, and the measures are contaminated with i.i.d Gaussian noise.

This method however has a number of drawbacks:

- As such, the solution f is not represented in a very compact way. If the measuring points are uniformly spread, the coefficients c_n are likely to be all of the same magnitude. The complexity of the representation thus increases exactly as the number of submitted samples. The evaluation cost also increases with time in the same proportion because the vector $[K(x, x_n)]_{n=1...N}$ is not sparse.
- the regularization model (and the underlying statistical hypothesis) have a strong influence on the shape of the solution. If $\mathcal{H}$ is for example a Sobolev space H^s, the convergence rate of this approach is of order s if the function is in H^s, and no

convergence can be proved otherwise. We thus have to predict as accurately as possible the real smoothness order of the unknown function. This for example does not allow for approximation rates that locally depend on the function smoothness.

Recent work by Vapnik and Chervonenkis provided a considerable enhancement to the radial basis function approach. They allow to select a small subset of basis functions that best represents the unknown function. These methods are built on the basis of theoretic results by Vapnik and Chervonenkis that have exhibited an explicit estimation of the generalization ability of a given set of approximation functions.

On the other hand, recent work by Beatson and Newsam, based on techniques developed by Greengard and Rokhlin have made a significant speedup possible in the evaluation and the fitting of radial basis function expansions, with multipole expansions.

However, radial basis function expansions suffer from intrinsic drawbacks that even these latest refinements do not solve, like a lack of adaptivity because of the requirement of an a priori uniform smoothness estimation of the unknown function.

Irregular grid techniques

Triangulation methods are also used to build piecewise linear interpolations of functions on the basis of pointwise samples. Nira Dyn *et al.* have developed triangulation methods that adapt to the shape of the function in order to minimize the estimation error for a given number of triangular patches.

Daubechies, Guskov and Sweldens have developed irregular grid techniques for an interpolation order larger than 2.

The main drawback of these approaches is that they become very complex in higher dimensions, and that the stability requirements on the grid geometry also become more complex when the approximation order increases.

Neural networks

A neural network like a multilayer perceptron is able to represent a parameterized class of functions $(f_\alpha)_{\alpha \in A}$ that can be built by combination of elementary functions. The parameters α are usually synaptic coefficients that have to be tuned. This tuning is done with a gradient method

$$\alpha_{n+1} = \alpha_n + (y_n - f(x_n)) \frac{\partial f}{\partial \alpha}(x_n)$$

Since the dependency of the output $f_\alpha(x)$ on these synaptic parameters α is complex and nonlinear, is usually difficult to guarantee that such a gradient scheme converges.

Single layer neural networks have been well studied. Density results (Cybenko), or of approximation order (Barron) have been proved. Variants with wavelet transfer functions have been suggested by Pati and Krishnaprasad, Zhang and Benveniste and Candès and Donoho and are described in more detail in Chap. 7.

1.2.1 Choice of a representation and nonlinear approximation

Our approach, which is detailed in Chap. 8, is based on the choice of a representation for the estimated functions. The functions we choose are finite linear combinations of functions of a fixed basis $\mathcal{G}$. This approach is with this respect comparable to the radial basis function method, and strongly differs from multilayer neural networks.

The choice of the family of basis functions $\mathcal{G}$ is a crucial point and is connected to the a priori assumption we make on the function : this family must approximate well (with a small number of coefficients) a function from a given smoothness class in which the unknown function is supposed to live.

Approximation with subsets of a basis can be done in two ways: linear and nonlinear approximation.

Linear approximation Regularization methods are based on the assumption that the function is in some Sobolev space. Functions in a Sobolev space $W^s([0, 2\pi])$ are well approximated with truncated Fourier expansions. If we define the Fourier functions

$$g_m(x) = e^{imx}$$

a good approximation of a function in W^s with $2N + 1$ coefficients is

$$f_{2N+1} = \sum_{m=-N}^{N} \langle f, g_m \rangle g_m$$

If the coefficients decay in m^{-s}, i.e.

$$|\langle f, g_m \rangle| \leq A|m|^{-s}$$

which is the case whenever $f \in W^s$, then the $2N + 1$ term approximation error has the following decay rate

$$\|f_N - f\|_{L_2} = \mathcal{O}(N^{1/2-s})$$

The Fourier functions are thus good basis functions to represent function of uniform smoothness (Sobolev smoothness), regardless of the smoothness index s of the functions to approximate.

This $2N + 1$ term approximation is a linear approximation, because the expansion is always truncated to the same subset of coefficients $\{m : |m| \leq N\}$, and the approximation is thus only the projection of the original function on a fixed subspace.

Nonlinear approximation Another approach consists in choosing adaptively N coefficients depending on the function f to approximate. The approximation f_N is not a linear function of f any more, because it is projected on a subspace that depends on f. This is therefore called nonlinear approximation. If the family $\mathcal{G}$ is an orthonormal basis, the best approximation in L_2 norm of f in $\mathcal{G}$ is obtained by simply selecting the N highest coefficients in the expansion of f in that basis.

Let us now assume that $\mathcal{G}$ is a Hilbert basis. Functions that are well approximated in this way are functions whose expansion over the basis $\mathcal{G}$ is sparse, which means that a small number of coefficients are large, and most of the coefficients are very small. This consists in saying if the basis functions g_m of $\mathcal{G}$ are reordered into (g_{m_k}) so that the sequence $|\langle f, g_{m_k}\rangle|$ is nonincreasing, then the decay of the rearranged coefficients must be:

$$|\langle f, g_{m_k}\rangle| \leq Ak^{-s} \tag{1.1}$$

In this case, the N term nonlinear approximation error decay is

$$\left\| f - \sum_{k=1}^{N} \langle f, g_{m_k}\rangle g_{m_k} \right\|_{L_2} = \mathcal{O}(N^{1/2-s})$$

We can show that if for a given function f, the nonlinear approximation error decays with this rate, then its rearranged coefficients also fulfill (1.1).

Nonlinear approximation also allows to approximate efficiently classes of functions that are larger than for linear approximation. If we take an orthonormal wavelet basis

$$\psi^n_{j\boldsymbol{k}} = 2^{dj/2}\psi^n(2^j\boldsymbol{x} - \boldsymbol{k})$$

the classes of functions that are well approximated in such bases with nonlinear approximation are Besov classes. These classes are richer than the corresponding Sobolev classes, and contain functions that have some singularities.

Tree structure A nonlinear approximation cannot in practice be performed in its most general setting, because it is not possible to scan an infinite number of coefficients to find the largest ones, and also because coding a nonlinear approximation includes the coding of the selected subset of basis functions, and coding any arbitrary subset of a countable set can be arbitrarily expensive.

For this reason, it is customary to add a structure to these coefficient subsets, like requiring that they make up a subtree of the wavelet coefficient tree. Wavelets have a standard tree structure in which a d–variate wavelet at resolution j has 2^d different sons at the next resolution that are at a similar location.

This construction is motivated by the following considerations

- for piecewise smooth functions f, a wavelet coefficient is usually large only if the parent wavelet coefficient is also large.
- according to a theorem proved by Cohen, DeVore *et al.*, constraining the choice of a coefficient subset to be on a subtree of the wavelet coefficient tree is of little cost. The new approximation class in included in Besov spaces of indices arbitrarily close to that of the original Besov space.

1.2.2 Suggested method for building an interpolant

The method we suggest for building an interpolant $f_{\mathcal{X}}$ on the basis of the samples $(\boldsymbol{x}, f(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$ is the following. We write $f_{\mathcal{X}}$ as a linear combination of a finite wavelet

subfamily of the basis that is a subtree (or has similar properties):

$$f_{\mathcal{X}} = \sum_{(j,\boldsymbol{k},s)\in T} c^s_{j\boldsymbol{k}} \psi^s_{j\boldsymbol{k}}$$

The wavelet coefficients are then computed by inverting a system of selected interpolation constraints in the subspace spanned by this subfamily:

$$\sum_{(j,\boldsymbol{k},n)\in T} c^n_{j\boldsymbol{k}} \psi^n_{j\boldsymbol{k}}(x_m) = y_m \qquad\qquad m = 1\dots$$

We describe three methods that can all be split into three steps.

Allocation The first step consists in choosing a subfamily of the wavelet basis by iteratively building a mapping from the measures $\boldsymbol{x} \in \mathcal{X}$ to the wavelet basis. This provides a set of n wavelets if we have n measures. The technique used for this is called *allocation*.

Stability control The second step is supposed to modify this initial choice in order to obtain a stable system of interpolation constraints. This can be done in several ways. In Chap. 8, we describe a selection rule of wavelet/mesure pairs based on a *good relative positioning* (GRP) criterion. Later in the next chapter, we describe other approaches of *a posteriori stability control* and of *local regularization*.

System solving The third step simply consists in solving the resulting linear system.

The first step we now describe consists in choosing wavelets on the basis of measuring points $\boldsymbol{x} \in \mathcal{X}$ by relying exclusively on a geometric criterion (of relative locations of the measuring points and the wavelet centers). We call this "allocation" because it is based on the construction of a 1-1 mapping between the measures and the selected wavelet subfamily.

Allocation

The allocation method is designed to select N wavelets depending on a set of N measuring points. It consists in placing these measuring points into the wavelet tree in a tree descent which is guided by the relative positions of the wavelets and the measuring point to be placed in the tree.

This choice is motivated in several ways. First, we favor in this way the placing of each measure as close as possible to the corresponding wavelet. Moreover, high resolution wavelets will be used only at location where a high number of measures is provided, which seems sensible.

The exact definition of this scheme and its properties (existence of an optimality criterion to define a best and almost unique allocation, etc.) are given in Sec. 8.3.

Stability control with a good relative positioning criterion: convergence results

With such an allocation scheme, we can now build a square linear system of N linear constraints in a subspace spanned by N wavelets. However, nothing guarantees so far that the resulting linear system is invertible and stable.

We therefore describe a selection criterion that is purely geometric to select a subset of the resulting wavelet/measure pairs: the good relative positioning (GRP) criterion. This selection can be done in very short time ($\mathcal{O}(N)$).

The algorithm consisting of the three above steps can then be proved to be converging. We can also provide estimations of the convergence rate of this method. Defining the fill distance h of a set of measure points $\mathcal{X}$ in a bounded domain D as

$$h = \max_{x \in D} \min_{n=1\ldots N} |x_n - x|$$

we are then able to prove the following results.

Stability The interpolant $f_{\mathcal{X}}$ obtained by interpolation with the samples $(\boldsymbol{x}, f(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$ fulfills the following stability inequality:

$$\|f_{\mathcal{X}}\|_\infty \leq M \|f\|_\infty$$

This stability result is the key result of the analysis of this approach. The convergence rate estimations are then a relatively direct consequence of this result.

Convergence rates If a function f is uniformly Lipschitz–α and if the wavelets used have β vanishing moments, we have the following error bound:

$$\|f_{\mathcal{X}} - f\|_\infty \leq M h^{\min(\alpha,\beta)}$$

We can even show that if f is only continuous, then we have

$$\|f_{\mathcal{X}} - f\|_\infty \to 0 \qquad \text{as } h \to 0$$

This interpolation method does not fail when the function f is not very smooth. This differs from regularization. For the latter, if β denotes the approximation order of the regularization scheme, the convergence only takes place if $\beta < \alpha$.

Good conditioning We can define the separation q distance of the set of measuring points $\mathcal{X}$ as

$$q = \min_{\substack{\boldsymbol{x},\boldsymbol{x}'\in\mathcal{X} \\ \boldsymbol{x}\neq\boldsymbol{x}'}} |\boldsymbol{x} - \boldsymbol{x}'|$$

Then, the norm of the system inverse matrix is bounded by some multiple of $|\log q|$ while for regularization, the only bound available is $q^{-\beta}$, and the collocation matrix has thus a worse condition number. We must also note that in our case, the system matrix is sparse.

Adaptive approximation If the function f has a non uniform smoothness, and is for example uniformly α–Lipschitz except on the support of a singularity S on a vicinity of which it is uniformly Lipschitz–β, with $\beta < \alpha$, then we have the following local error bound

$$|f_{\mathcal{X}}(\boldsymbol{x}) - f(\boldsymbol{x})| \leq M h^\alpha + M' h^\beta e^{-M'' d(\boldsymbol{x},S)/h}$$

where $d(\boldsymbol{x}, S)$ is the distance from $\boldsymbol{x}$ to the set S. This means that the bad influence of a singularity on the convergence rate is only local, and has an intensity that is exponentially decaying with the distance to the singularity. This decay is all the faster as h is small.

Incremental approaches

In certain settings, we may want to evaluate the estimated function and to refine the function estimation at the same time. This happens in real time applications, like in control.

The allocation algorithm is incremental, and the system matrix and its inverse are also sparse. Updating these matrices when a new measure is provided can be done with low complexity computations. The computational cost at each iteration is bounded by some multiple of $(\log N)^2$, where N is the number of measures already provided.

1.2.3 A posteriori stability control and local regularization

The major drawback of the approach based on the GRP criterion is that some measures are rejected by the algorithm. The part of rejected measures increases significantly with the input space dimension. Moreover, if the measuring points are located on a submanifold of the input space, the GRP criterion discards all the provided measures, so no convergence can be proved nor obtained numerically. For this reason, we have started to design two other approaches where the stability criterion is not the GRP criterion any more. The first one consists in *controlling a posteriori the stability* of the system, and the second approach consists in doing *local regularization*.

Together with the incremental implementation of the interpolation method base on the GRP stability control, these two variants are described in the prospective chapter 9.1. Numerical experiments are also shown. The stability is up to now insufficient, but we indicate in this chapter how to remedy these drawbacks in a near future.

Chapter 2

Wavelets

Abstract

The purpose of this chapter is to give a short introduction to the wavelets we use later in this work. We describe them in the more general framework of time-frquency analysis, and remind some of their main properties.

2.1 Local frequency

Wavelets have been introduced about 50 years ago to fill a gap between two extreme ways of representing a signal: its time representation which can be seen as an expansion over the Dirac masses:

$$f(t) = \int_{\mathbb{R}} f(u)\delta(t-u)du$$

and its frequency representation or Fourier transform, which is its expansion over the complex exponentials:

$$f(t) = \int_{\mathbb{R}} \hat{f}(\omega)e^{i\omega t}d\omega$$

The first representation gives an information of maximal time resolution: the value $f(t)$ represents a signal intensity at time t. No frequency information is available. A single pointwise value does not provide any information on the frequency content of f. Conversely, the Fourier representation gives an accurate frequency information, but no time information at all. To give a musical analogy, let us suppose that f represents the time curve of a tune. The time representation of f allows to know when we have notes and when we have rests, but not the pitch. On the other hand, the frequency representation tells what pitches can be heard in the tune, but not when exactly they can be heard.

Each of these representations contains the whole information on the signal, because the Fourier transform is a bijective mapping from one representation to the other. However, in each case, only one kind of information is explicitly available.

We can restate the above observation by saying that the Dirac distributions $\delta(.-t)$ have an infinitely high spatial resolution, but no frequency resolution at all, and the converse for the complex exponentials $t \mapsto e^{i\omega t}$. The question was then to find a way of representing signals between these extremes, in which a mixed information is explicitly given, like "there we hear an A, and there a $B^\flat$". Morlet and Gabor suggested to use basis functions that are half way between Dirac masses and complex exponentials, that are both localized in time and in frequency.

A theoretic limit to this is well known: the Heisenberg inequality. Let f be a square integrable function, of L_2 norm equal to 1:

$$\int |f(t)|^2 dt = 1\,.$$

We define the center $c(f)$ and the width $\Delta(f)$ of such a function as

$$c(f) = \int t|f(t)|^2 dt$$
$$\Delta(f) = \sqrt{\int (t-c(f))^2 |f(t)|^2 dt}$$

The Heisenberg inequality is a fundamental inequality that holds for any function f of norm 1:

$$\Delta(f)\Delta(\hat{f}) \geq \frac{1}{2} \tag{H}$$

If we call the frequency width of a function f the width of its Fourier transform $\hat{f}$, this inequality implies that it is not possible to find a function that has a width and a frequency width both arbitrarily small.

We also know the functions that achieve an optimal trade-off: translated and modulated Gaussian functions:

$$Ae^{-(t-t_0)^2/2\Delta t^2} e^{i\omega_0 t}$$

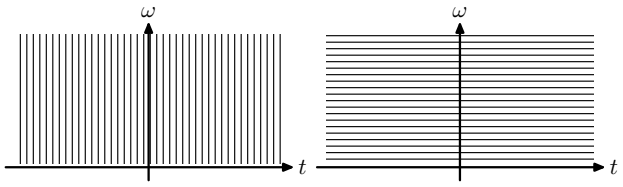
where A is a normalization coefficient such that the above function has an L_2 norm equal to 1. For such functions, and only for them, the Heisenberg inequality is an equality. These functions are now called Gabor wavelets.

2.2 Time-frequency and time-scale representations

To such a function, we associate a time-frequency rectangle on the (t,ω) plane centered around $(c(f), c(\hat{f}))$ and of dimensions $\Delta(f) \times \Delta(\hat{f})$. This rectangle is an intuitive representation of a time-frequency support of a function. To a basis of $L_2(\mathbb{R})$, we also associate a paving of the time-frequency plane, by such rectangles. The rectangles in this paving are centered around the time-frequency center $(c(f), c(\hat{f}))$ of the functions, and their dimensions are proportional to the time frequency dimensions $\Delta(f) \times \Delta(\hat{f})$ in order to draw a partition

of the time-frequency plane. This representation is a little arbitrary, as no theoretical result relates a time-frequency paving to the fact that the corresponding function family is a basis.

The time-frequency paving for Dirac functions and complex exponentials are paving with infinitely thin and long rectangle as represented in Fig. 2.1.



(a) Dirac (b) Fourier

Figure 2.1: Time-frequency paving for Dirac and Fourier representations

To represent a function as a linear combination of such functions, using all possible combinations of the parameters t_0, ω_0 and Δt is overly redundant. Two different approaches have been suggested:

- a time-frequency approach, that consists in choosing the time width of the functions g independently of the modulation frequency. Such functions can be written as

$$g_{t_0,\omega_0}(t) = e^{i\omega_0 t} g_0(t - t_0)$$

  where $g_0(t) = A_0 e^{-t^2/2\Delta t^2}$. Such a representation is called windowed Fourier transform

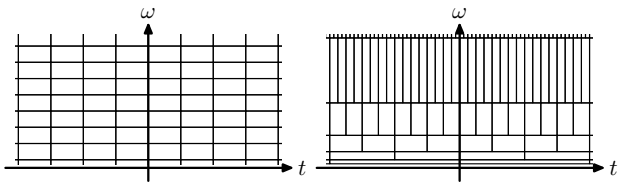
- a time-scale approach, in which the width of the functions is inverse proportional to the frequency (the product $\omega_0 \Delta t$ is a constant equal to $c0$. We then obtain the following parameterized functions:

$$g_{t_0,\Delta t}(t) = \frac{1}{\sqrt{\Delta t}} g_0\left(\frac{t - t_0}{\Delta t}\right)$$

  where $g_0(t) = A_0 e^{-t^2/2\Delta t^2} e^{ict}$.

The time frequency pavings illustrate the differences between both approaches. In the case of time-frequency analysis, the paving is that of rectangles that are all translates of the same one. In case of time-scale analysis, the rectangles all have the same area, but their *relative* frequency resolution $\Delta\omega/\omega_0$ is constant.

The wavelets we use in this work are time-scale wavelets. The advantages of a time-scale approach are numerous. We dispose of efficient ways to design such family for which the related computation are very fast. The corresponding bases have a very simple form, since all functions have the same shape, and last they have a spatial width which is always proportional to the spatial resolution related to their frequency bandwidth through the Heisenberg inequality.



(a) Time-frequency (b) Time-scale

Figure 2.2: Pavings of the time-frequency plane for the time-frequency representation and the time-scale representation

2.3 Continuous wavelet transform

We choose a basis function ψ called wavelet that fulfills an admissibility condition:

$$C_\psi = \int_{\mathbb{R}} \frac{|\hat{\psi}(\omega)|^2}{\omega} d\omega < +\infty \tag{2.1}$$

We notice that if the Fourier transform of this function is continuous, it has to vanish on 0, which is not the case of the Gabor function above. In practice, we can however consider that the Gabor function satisfies this assumption since for the chosen parameters, the value of its Fourier transform on 0 is very small (but not zero).

We define the continuous wavelet transform $\mathcal{W}f$ of the function f with the following formula:

$$\mathcal{W}f(t,s) = \int_{\mathbb{R}} f(\tau) \frac{1}{\sqrt{s}} \overline{\psi\left(\frac{\tau - t}{s}\right)} d\tau \tag{2.2}$$

The inverse transform is

$$f(t) = \frac{1}{C_\psi} \iint_{\mathbb{R}^2} \mathcal{W}(\tau, s) \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right) d\tau \frac{ds}{s^2} \tag{2.3}$$

The same way as for the Fourier transform, this transformation is quasi-isometric:

$$\|f\|^2_{L_2(\mathbb{R})} = \frac{1}{C_\psi} \iint_{\mathbb{R}^2} |\mathcal{W}f(t,s)|^2 dt \frac{ds}{s^2}. \tag{2.4}$$

However, the continuous wavelet transform $\mathcal{W}f$ of a function f is a very redundant representation. $\mathcal{W}f$ fulfills a reproducing kernel equation:

$$\mathcal{W}f(t,s) = \frac{1}{C_\psi} \iint_{\mathbb{R}^2} \mathcal{W}(\tau,\sigma) K(\tau,t,\sigma,s) dt \frac{ds}{s^2}$$

with a kernel K defined as

$$K(\tau,t,\sigma,s) = \int_{\mathbb{R}} \frac{1}{\sqrt{\sigma s}} \psi\left(\frac{t'-t}{s}\right) \overline{\psi\left(\frac{t'-\tau}{\sigma}\right)} dt'$$

2.4 The discrete wavelet transform

Morlet suggested to build bases or frames built according to the following scheme:

$$g_{t_0,\Delta t}(t) = \frac{1}{\sqrt{\Delta t}} g\left(\frac{t - t_0}{\Delta t}\right)$$

where the values of Δt are chosen on a geometric scale, and where the translation steps are proportional to Δt :

$$\Delta t = b^j$$
$$t_0 = k\Delta t$$

A widely used set of scales are dyadic scales 2^j, and we obtain families of functions of the form $g_0(2^j(t - 2^{-j}k)) = g(2^j t - k)$ where j and k are integers. The most common normalization being an L_2 normalization, we obtain families of the form $(\psi_{jk})_{j,k\in\mathbb{Z}}$ where $\psi_{jk}(t) = 2^{j/2}\psi(2^j t - k)$.

In a paper written in 1987, Stéphane Mallat has drawn a parallel between the time-scale representations of Morlet and the quadrature mirror filters Burt, Adelson and Simoncelli used in image compression.

He described a family of wavelet decompositions that can be implemented very efficiently with a "fast wavelet transform", for which the wavelet ψ is an infinite convolution of discrete filters. More precisely, we can find discrete filters m_0 and m_1:

$$k \mapsto m_0[k] \qquad k \in \mathbb{Z}, \tag{2.5}$$
$$k \mapsto m_1[k] \qquad k \in \mathbb{Z}; \tag{2.6}$$

whose Fourier transforms $\omega \mapsto m_0(\omega)$ and $\omega \mapsto m_1(\omega)$ are 2π–periodic functions. We assume that there exists a scaling function ϕ and a wavelet ψ in $L_2(\mathbb{R})$ such that

$$\hat{\phi}(\omega) = \prod_{k=1}^{+\infty} m_0\left(\frac{\omega}{2^k}\right) \tag{2.7}$$
$$\hat{\psi}(\omega) = m_1\left(\frac{\omega}{2}\right)\hat{\phi}\left(\frac{\omega}{2}\right) \tag{2.8}$$

Under some conditions on m_0 and m_1, the family (ψ_{jk}) is an orthonormal basis, and the wavelet decomposition of a sampled signal can be done in a short time with a sequence of filtering and subsampling steps.

This approach reduces significantly the complexity of the object to design. Instead of choosing a function, we choose the discrete (and most usually finite) set of coefficients of the filters.

2.5 Multiresolution analyses

The theory and the tools for the wavelets that can be designed from discrete filters as been widely developed in this last decade, and there exist a number of theorems relating the

properties of the discrete filters with those of the resulting wavelets. Moreover, a number of wavelet families have been constructed that carry either the name of some property they have, or the name of their inventor.

2.5.1 Theoretic framework

The theoretic framework developed by Stéphane Mallat is based on the definition of *multiresolution analyses*. A multiresolution analysis is a sequence of closed subspaces of $L_2(\mathbb{R})$ denoted by $(V_j)_{j\in\mathbb{Z}}$, which has the following properties:

$$V_j = \left\{ \sum_{k\in\mathbb{Z}} a_k \phi_{jk} : a_k \in \mathbb{R} \right\} \qquad \text{(Riesz space)} \tag{2.9a}$$

$$V_j \subset V_{j+1} \tag{2.9b}$$

$$\bigcap_{j\in\mathbb{Z}} V_j = \{0\} \tag{2.9c}$$

$$\overline{\bigcup_{j\in\mathbb{Z}} V_j} = L_2(\mathbb{R}) \tag{2.9d}$$

We can make the following remarks:

- The assumption (2.9a) means that V_j is a Riesz space spanned by the family $(\phi_{jk})_{k\in\mathbb{Z}}$. The definition depends on the underlying topology for the functional space, because it can be defined as the closure of the space of finite linear combinations of ϕ_{jk}. This also imposes a constraint on the function ϕ. For an L_2 metric, the mapping

$$\begin{aligned} \ell_2(\mathbb{Z}) &\to L_2(\mathbb{R}) \\ (a_k)_{k\in\mathbb{Z}} &\mapsto \sum_{k\in\mathbb{Z}} a_k \phi_{0k} \end{aligned}$$

  has to be continuous. A function ϕ with a too slow decay at infinity cannot be used.

- Intuitively, we can consider that the set of functions V_{j+1} is a "richer" or "denser" set as V_j, which does not means that the latter is a subset of the former. However, equation (2.9b) assumes this.

  Because of scale and translation invariance of the above definitions, we can check that this last assumption is equivalent to assuming that $\phi \in V_1$, which means that there exists a sequence of coefficients $(m_0[k])_{k\in\mathbb{Z}}$ such that

$$\phi(t) = 2\sum_{k\in\mathbb{Z}} m_0[k]\phi(2t-k) \tag{2.10}$$

  This is how the discrete filter m_0 appears.

- The assumption (2.9c) is stated more or less for clarity, because it always fulfilled whenever ϕ is not 0. The assumption (2.9d) is also fulfilled whenever $\hat{\phi}$ does vanish on 0.

Wavelets then appear as a natural way to express the difference between two successive spaces V_j and V_{j+1}. We build for this a Riesz space W_0 such that:

$$V_0 \oplus W_0 = V_1 \tag{2.11}$$

The subspace W_0 is spanned by translates of a function ψ:

$$W_0 = \left\{ t \mapsto \sum_{k\in\mathbb{R}} d_k \psi(t-k) : a_k \in \mathbb{R} \right\}$$

This imposes that the function ψ be in the space V_1, thus again:

$$\psi(t) = \sum_{k\in\mathbb{Z}} m_1[k]\phi(2t-k) \tag{2.12}$$

Hence the second discrete filter m_1.

We show that the functions ϕ and ψ are then defined by the sole choice of these two filters: m_0 and m_1. The formulae are

$$\hat{\phi}(\omega) = \prod_{k=1}^{+\infty} m_0\left(\frac{\omega}{2^k}\right) \qquad \text{par itération de (2.10),} \tag{2.13a}$$

$$\hat{\psi}(\omega) = m_1\left(\frac{\omega}{2}\right)\hat{\phi}\left(\frac{\omega}{2}\right) \qquad \text{par (2.12).} \tag{2.13b}$$

2.5.2 Wavelet bases

The equation (2.11) can be transposed at any scale j:

$$V_j \oplus W_j = V_{j+1} \tag{2.14}$$

and by iterating, we get:

$$V_j \oplus W_j \oplus \cdots \oplus W_{j'-1} = W_{j'} \qquad \text{if } j < j' \tag{2.15}$$

If j' goes to $+\infty$ (and resp. also j to $-\infty$), we obtain two decompositions:

$$L_2(\mathbb{R}) = V_j \oplus \overline{\bigoplus_{j'=j}^{+\infty} W_{j'}} \qquad \text{for all } j \in \mathbb{Z} \tag{2.16}$$

$$L_2(\mathbb{R}) = \overline{\bigoplus_{j'=-\infty}^{+\infty} W_{j'}} \tag{2.17}$$

The union of all Riesz bases in each of these direct sums thus provides several wavelet bases:

$$\mathcal{B}_j = \{\phi_{jk} : k \in \mathbb{Z}\} \cup \{\psi_{j'k} : j' \geq j,\, k \in \mathbb{Z}\} \tag{2.18}$$

$$\mathcal{B} = \{\psi_{jk} : j \in \mathbb{Z},\, k \in \mathbb{Z}\} \tag{2.19}$$

2.5.3 Wavelet transform

The wavelet transform is used for sampled signals. Most of the time, this sampling is written as

$$f = \sum_{k\in\mathbb{Z}} 2^{j/2} f[k/2^j]\phi_{jk}$$

where the sample $f[k/2^j]$ is estimated with:

$$f[k/2^j] \simeq f(k/2^j)$$

The signal to be transformed is thus supposed to be expanded over the Riesz basis of V_j. The transformation consists in finding its expansion over the basis corresponding to the direct sum:

$$V_L \oplus W_L \oplus W_{L+1} \oplus \cdots \oplus W_{j-1}$$

The transformation is recursive and consists in replacing the representation of a component expanded over some $V_{j'}$ with a representation over $V_{j'-1} \oplus W_{j'-1}$. We successively obtain decompositions adapted to the following direct sums:

$$\begin{gathered} V_{j-1} \oplus W_{j-1} \\ V_{j-2} \oplus W_{j-2} \oplus W_{j-1} \\ \vdots \\ V_L \oplus W_L \oplus W_{L+1} \oplus \cdots \oplus W_{j-1} \end{gathered}$$

2.5.4 Dual filters, dual wavelets

The fundamental step in a wavelet transform is therefore the following basis change:

$$V_{j+1} \to V_j \oplus W_j$$

It can be written as a mapping

$$\begin{gathered} \ell_2(\mathbb{Z}) \to \ell_2(\mathbb{Z}) \times \ell_2(\mathbb{Z}) \\ (a_{j+1,k})_{k\in\mathbb{Z}} \mapsto [(a_{jk})_{k\in\mathbb{Z}}, (d_{jk})_{k\in\mathbb{Z}}] \end{gathered}$$

If we denote by A_j and D_j the 2π–periodic functions whose coefficients are the discrete sequences $k \mapsto a_{jk}$ and $k \mapsto d_{jk}$:

$$\begin{aligned} A_j(\omega) &= \sum_{k\in\mathbb{Z}} a_{jk} e^{-ik\omega} \\ D_j(\omega) &= \sum_{k\in\mathbb{Z}} d_{jk} e^{-ik\omega} \end{aligned}$$

this fundamental step can be written as a multiplication with a transfer matrix:

$$\begin{bmatrix} A_j(2\omega) \\ D_j(2\omega) \end{bmatrix} = \begin{bmatrix} m_0(\omega) & m_0(\omega+\pi) \\ m_1(\omega) & m_1(\omega+\pi) \end{bmatrix} \begin{bmatrix} A_{j+1}(\omega) \\ A_{j+1}(\omega+\pi) \end{bmatrix}$$

A necessary condition for such a transformation to be invertible in $L_2([0,2\pi])^2$ is that the transfer matrix

$$T(\omega) = \begin{bmatrix} m_0(\omega) & m_0(\omega+\pi) \\ m_1(\omega) & m_1(\omega+\pi) \end{bmatrix}$$

be bounded and of bounded inverse uniformly on $[0,2\pi]$. In this case, the dual transfer matrix is $\tilde{T}(\omega) = T(\omega)^{-T}$. There exist two other 2π–periodic functions $\tilde{m}_0$ and $\tilde{m}_1$ such that $\tilde{T}(\omega)$ is written:

$$\tilde{T}(\omega) = \begin{bmatrix} \tilde{m}_0(\omega) & \tilde{m}_0(\omega+\pi) \\ \tilde{m}_1(\omega) & \tilde{m}_1(\omega+\pi) \end{bmatrix}$$

These filters define dual wavelets $\tilde{\phi}$ and $\tilde{\psi}$ with equations similar to (2.13a) and (2.13b):

$$\hat{\tilde{\phi}}(\omega) = \prod_{k=1}^{+\infty} \tilde{m}_0\left(\frac{\omega}{2^k}\right) \tag{2.20a}$$

$$\hat{\tilde{\psi}}(\omega) = \tilde{m}_1\left(\frac{\omega}{2}\right) \hat{\tilde{\phi}}\left(\frac{\omega}{2}\right) \tag{2.20b}$$

The functions $\tilde{\phi}$ and $\tilde{\psi}$ are dual wavelets in the sense that for all j, we have the expansions on $L_2(\mathbb{R})$:

$$f = \sum_{k\in\mathbb{Z}} \left\langle f, \tilde{\phi}_{jk} \right\rangle \phi_{jk} + \sum_{j'\geq j, k\in\mathbb{Z}} \left\langle f, \tilde{\psi}_{j'k} \right\rangle \psi_{j'k} \tag{2.21}$$

for all $j \in \mathbb{Z}$ and $f \in L_2(\mathbb{Z})$, and with j going to $-\infty$, we obtain:

$$f = \sum_{j,k\in\mathbb{Z}} \left\langle f, \tilde{\psi}_{jk} \right\rangle \psi_{jk} \tag{2.22}$$

2.5.5 The fast wavelet transform

The coefficients of the filters m_0 and m_1, and of the dual filters $\tilde{m}_0$ and $\tilde{m}_1$ are used in the basis change

$$\{\phi_{jk} : k \in \mathbb{Z}\} \cup \{\psi_{jk} : k \in \mathbb{Z}\} \leftrightarrow \{\phi_{j+1,k} : k \in \mathbb{Z}\}$$

with the following formulae:

- for the forward wavelet transform, we have

$$a_{jk} = 2\sum_{l\in\mathbb{Z}} \tilde{m}_0[k]a_{j+1,2l-k}$$
$$d_{jk} = 2\sum_{l\in\mathbb{Z}} \tilde{m}_1[k]a_{j+1,2l-k}$$

- and for the inverse wavelet transform, we get

$$a_{j+1,k} = \frac{1}{2}\sum_{l\in\mathbb{Z}} m_0[2l-k]a_{jl} + m_1[2l-k]d_{jl}$$

The cascade filtering schemes are displayed for the direct and the inverse transforms in Fig. 2.3 between resolutions $j = 0$ and $j = -3$.

For a finite number N of samples, a wavelet transform (until any resolution possible with such a number of samples) costs less that $A \times N$ flops, where the constant A depends on the chosen filters. This in theory better than a fast Fourier transform which costs $N \log N$ flops.

2.5.6 Orthogonal wavelets

Orthogonal wavelets are wavelets ψ such that the family $(t \mapsto 2^{j/2}\psi(2^j t - k))_{j,k\in\mathbb{Z}}$ is an orthonormal basis of $L_2(\mathbb{R})$. This means that $\phi = \tilde{\phi}$ and $\psi = \tilde{\psi}$, and is equivalent to saying that the transfer matrix and the dual transfer matrix are equal, or that the transfer matrix is unitary for all ω. This can be written in terms of the filters m_0 and m_1 as:

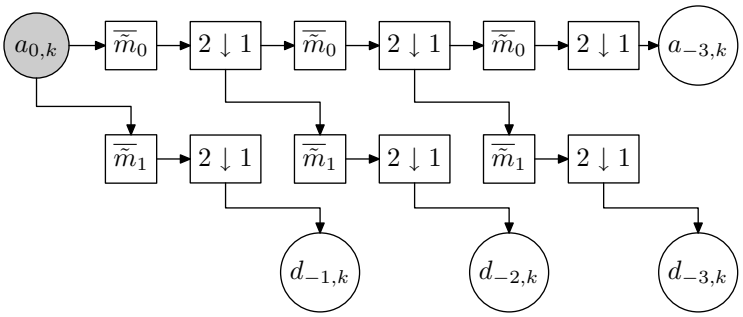
$$\begin{array}{rcrcll} |m_0(\omega)|^2 & + & |m_0(\omega+\pi)|^2 & = & 1 & \forall\omega \\ m_0(\omega)\overline{m_1(\omega)} & + & m_0(\omega+\pi)\overline{m_1(\omega+\pi)} & = & 0 & \forall\omega \\ |m_1(\omega)|^2 & + & |m_1(\omega+\pi)|^2 & = & 1 & \forall\omega \end{array}$$

In this case, m_0 and m_1 are called quadrature mirror filters, as Esteban and Galand called them, and after them Adelson and Simoncelli. Moreover, all the direct sums above are then orthogonal direct sums. In practice, once the filter m_0 is chosen, almost a single choice for m_1 is possible, which is

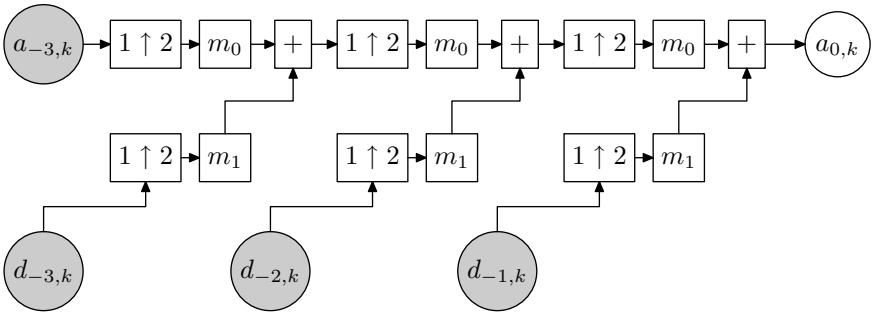
$$m_1(\omega) = e^{i\omega}\overline{m_0(\omega+\pi)}$$

Historically, the first wavelets to be defined in the multiresolution analysis framework were orthogonal wavelets (Meyer, Mallat), so that other wavelets were afterwards called biorthogonal wavelets. The prefix "bi" meaning that we need two wavelet bases instead of one. The first for the decomposition, and the second for the signal reconstruction. A systematic analysis of biorthogonal wavelets has been done by Cohen, Daubechies *et al.*

There exist a number of orthogonal wavelets that are frequently used. The most known such wavelets are Daubechies' wavelets. They achieve an optimal trade-off between support size and number of vanishing moments (two opposite constraints, in some way comparable



(a) Direct transform



(b) Inverse transform

Figure 2.3: Fast wavelet transform. The shaded circle represent the input, while the white circle represent the output.

to the Heisenberg inequality). There exist also many other wavelets, like the *coiflets* named after Ronald Coifman, or the *symmlets*, that are almost symmetric.

Orthogonal wavelet bases have a considerable theoretic advantage in compression or denoising applications : first the error metric is in general an L_2 metric, and this can be easily expressed in terms of orthogonal wavelet coefficients. In the case of denoising, a Gaussian white noise has expansion coefficients over an orthonormal basis that are i.i.d Gaussian variables.

In practice however, orthogonal wavelets do not have the same flexibility as that offered by the biorthogonal wavelets. One can for example show that they can never (up to some trivial cases) be symmetric. The dual relationship between dual wavelets for computing expansion coefficients $c_{jk} = \langle f, \tilde{\psi}_{jk} \rangle$ and wavelets ψ_{jk} for reconstruction $f = \sum c_{jk} \psi_{jk}$ can be reversed, to get:

$$f = \sum_{j,k \in \mathbb{Z}} \langle \psi_{jk}, f \rangle \tilde{\psi}_{jk}$$

that can be compared to (2.22).

2.6 Coefficient decay, smoothness and approximation rates

If we want subexpansions of a function to converge rapidly to the original function, the expansion coefficients have to decay rapidly as $j \to +\infty$. This decay is connected to the number of *vanishing moments* of the dual wavelet and to the smoothness of the function.

We say that $\tilde{\psi}$ has p vanishing moments if

$$\int_{\mathbb{R}} \tilde{\phi}(t) t^k dt = 0$$

for all k in $\{0, \ldots, p-1\}$. This is the same as assuming that the Fourier transform of $\tilde{\psi}$ has a zero of order p in $\omega = 0$, or as assuming that ψ is orthogonal to any polynomial of degree less that p.

We can show that if f is p times differentiable with a bounded derivative of order p over an interval I, its wavelet coefficients decay in $2^{-j(p+1/2)}$ on I, i.e. there exists a bound M such that

$$\left| \langle \tilde{\psi}_{jk}, f \rangle \right| \leq M 2^{-(p+1/2)j} \qquad \text{if } \operatorname{supp} \tilde{\psi}_{jk} \subset I$$

To show this bound, we write a Taylor expansion of f around the center of the wavelet $\tilde{\psi}_{jk}$. If $u = k/2^j$, we have:

$$f(t) = \sum_{k=0}^{p-1} \frac{(t-u)^p}{p!} f^{(p)}(u) + (t-u)^p r(t)$$

where the function $r(t)$ is bounded by the p^{th} derivative of f. By inner product of f with $\tilde{\psi}_{jk}$, the sum of polynomial terms vanishes, and we have the following term left

$$\begin{aligned}\langle \tilde{\psi}_{jk}, f\rangle &= \int (t-u)^p r(t)\overline{\psi_{jk}(t)}dt \\ &= \int t^p r(t+u)\overline{\psi_{j0}(t)}dt\end{aligned}$$

which can be bounded with a simple variable change by $M2^{-(p+1/2)j}$. We thus see that the local regularity of the function is strongly connected to the wavelet coefficient decay. A quasi converse statement is also true: if the coefficient decay is $\langle f, \tilde{\psi}_j\rangle \leq M2^{-(p+1/2)j}$ and if ψ is p–Lipschitz, then f is r–Lipschitz for any $r < p$.

To illustrate the relationship between smoothness and coefficient decay, an example of wavelet coefficient expansion is displayed in Fig. 2.4.

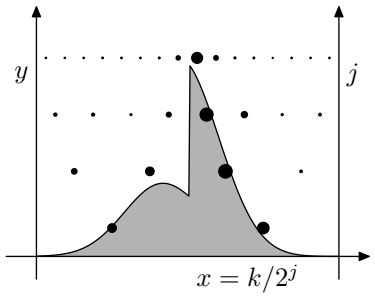


Figure 2.4: Wavelet coefficients of compactly supported function. Note that large coefficients at large j concentrate around singularities.

Remark

There is a strong connection between two dual parameters of a wavelet basis: (1) the number of vanishing moments of the dual wavelet, and (2) the smoothness of the wavelet. These parameters are involved respectively in the theorem estimating the wavelet coefficient decay from the function smoothness, and the converse theorem.

Yves Meyer has shown that if a wavelet is p times derivable, then the dual wavelet has at least $p+1$ vanishing moments. The converse is not true: the smoothness index of a wavelet is usually much lower than the number of vanishing moments of the dual wavelet.

For Daubechies wavelets, Cohen and Conze showed that the Lipschitz smoothness index α_N of a wavelet with N vanishing moments is asymptotically equal to:

$$\alpha \sim \left(1 - \frac{\log 3}{2\log 2}\right) N \simeq 0,2075\, N$$

2.7 Design of smooth wavelets and of the corresponding filters

Designing a wavelet family thus consists in choosing a pair of discrete filters (m_0, m_1). The corresponding transfer matrix

$$T(\omega) = \begin{bmatrix} m_0(\omega) & m_0(\omega+\pi) \\ m_1(\omega) & m_1(\omega+\pi) \end{bmatrix}$$

has to fulfill a perfect reconstruction condition, i.e. it must be bounded and of bounded inverse, which is relatively easy to check for compactly supported filters.

This is however not sufficient. Indeed, for some filters m_0 the associated function ϕ may not be in $L_2(\mathbb{R})$. If we simply take $m_0(\omega) = 1$, the corresponding scale function is a Dirac mass: $\delta(\omega)$. The generated functions also have to be smooth enough. In addition we must also require that the partial products

$$1_{[-2^j\pi, 2^j\pi]}(\omega) \prod_{k=1}^{j} m_0\left(\frac{\omega}{2^k}\right)$$

converge to $\hat{\phi}$ fast enough (at least in L_2 norm), so that the perfect reconstruction property of the filters can be effectively transposed into the biorthogonality of the wavelets:

$$\langle \phi_{jk}, \tilde{\phi}_{jk'} \rangle = \delta_{kk'}$$

This assumption is really necessary, as can be shown *a contrario* with the following example. The filter $m_0(\omega) = (1 + e^{2i\omega})/2$ is orthogonal, and the resulting scaling function is in $L_2(\mathbb{R})$

$$\phi(t) = \frac{1}{2} 1_{[0,2]}(t)$$

which is obviously not an orthogonal scaling function, since

$$\langle \phi_{jk}, \tilde{\phi}_{jk'} \rangle \neq \delta_{kk'} \qquad \text{for } |k - k'| = 1$$

To sum up, the three basic ingredients of an unconditional wavelet basis of L_2 built from filters are:

1. a pair of filters (m_0, m_1) fulfilling the perfect reconstruction assumption, and of dual filters $(\tilde{m}_0, \tilde{m}_1)$;

2. ϕ defined as

$$\hat{\phi}(\omega) = \prod_{k=1}^{+\infty} m_0\left(\frac{\omega}{2^k}\right)$$

should be in $L_2(\mathbb{R})$, as well as $\tilde{\phi}$ defined in a similar way with $\tilde{m}_0$;

3. the partial convolution products truncated on the frequency band $[-2^j\pi, 2^j\pi]$

$$1_{[-2^j\pi,2^j\pi]}(\omega)\prod_{k=1}^{j} m_0\left(\frac{\omega}{2^k}\right)$$

should converge to ϕ (and the same for $\tilde{\phi}$) in $L_2(\mathbb{R})$ norm, so that the filter duality is transposed into wavelet duality.

2.7.1 Sufficient conditions in the orthonormal case

A sufficient condition has been given by Mallat in his paper of 1987, for compactly supported orthogonal wavelets. He showed that an orthogonal filter m_0 (with $m_0(0) = 1$ and $|m_0(\omega)|^2 + |m_0(\omega+\pi)|^2 = 1 \forall \omega$) generates an orthonormal basis of $L_2(\mathbb{R})$ if

$$m_0(\omega) \neq 0 \qquad \text{for } \omega \in [-\pi/2, \pi/2]$$

This assumption has been relaxed afterwards, by imposing only that ϕ does not vanish on $[-\pi/3, \pi/3]$.

2.7.2 Sufficient condition for regularity

A theorem of Daubechies and Tchamitchian relates the smoothness of the scaling function ϕ to a factorization of the filter m_0.

Theorem 2.1 (Daubechies, Tchamitchian)

Let m_0 be a discrete filter such that the function

$$\omega \mapsto m_0(\omega)$$

is bounded, differentiable on 0 and such that $m_0(0) = 1$, and let ϕ be the function (tempered distribution) defined by

$$\hat{\phi}(\omega) = \prod_{k=1}^{+\infty} m_0\left(\frac{\omega}{2^k}\right)$$

We factor m_0 as

$$m_0(\omega) = \left(\frac{e^{i\omega}+1}{2}\right)^N r(\omega)$$

where N is as large as possible, and r is a bounded. We set

$$B_j = \sup_{\omega} \left| \prod_{k=1}^{j} r(2^k\omega) \right|$$

and

$$b_j = \frac{\log B_j}{j \log 2}$$
$$b = \inf_{j>0} b_j \qquad \textit{(critical exponent)}$$

Then the Fourier transform of ϕ is bounded by the following formula:

$$|\hat{\phi}(\omega)| \leq \frac{M}{1 + |\omega|^{N - b_j}} \qquad \forall j > 0$$

If we denote by G^α the Hölder space of exponent α, i.e. the set of functions f such that

$$\int |\hat{f}(\omega)|(1 + |\omega|)^\alpha d\omega < +\infty$$

ϕ is in H^α for any $\alpha < N - b_j - 1$.

Moreover, for any $\alpha < N - b_j - 1/2$, the function ϕ is in the Sobolev space of exponent α.

This an ingredient of the following theorem by Cohen

Theorem 2.2 (Cohen)

Let (m_0, m_1) and $(\tilde{m}_0, \tilde{m}_1)$ be two dual pairs of filters (thus perfect reconstruction filters). We denote by N, $\tilde{N}$, b and $\tilde{b}$ the factorization exponents and the critical exponents of m_0 and $\tilde{m}_0$. If $N - b > 1/2$ and $\tilde{N} - \tilde{b} > 1/2$, then the wavelets generated by these filters make up unconditional biorthogonal bases of $L_2(\mathbb{R})$.

Bibliography

[BCD+91] G. Beylkin, R. Coifman, I. Daubechies, S. G. Mallat, Y. Meyer, L. Raphael and M. B. Ruskai. *Wavelets*. Jones and Bartlett, Boston, 1991.

[Dau92] I. Daubechies. *Ten Lectures on Wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[Mal89a] S. G. Mallat. Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$. *Trans. Amer. Math. Soc.*, 315(1):69–87, 1989.

[Mal89b] S. G. Mallat. A theory for multiscale signal decomposition: The wavelet representation. *IEEE Trans. on Pattern and Machine Intelligence*, 11(7):674–693, 1989.

[Mal97] S. G. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1997.

Part I

Optic flow estimation

Chapter 3

Optic flow

Introduction

Optic flow measurement is an early vision processing step in computer vision, which is used in a wide variety of applications, ranging from three dimensional scene analysis to video compression and experimental physics.

The term optical flow was first used by the psychologist James Jerome Gibson in his study of human vision. In 1980–81, Horn and Schunck [HS80, HS81] devised a simple way to compute the optic flow based on regularization. This first work was then followed by a great number of contributions which proposed alternative methods. To mention a few, we can cite the spatiotemporal filtering methods, initiated by Adelson and Bergen [AB85], which split into energy based methods [Hee88] and phase based methods [FJ90], and region matching methods [BA83, BYX83, Ana89]. See Barron, Fleet and Beauchemin [BFB94] for an extensive survey of these methods [BFB94].

Many authors noticed that a good way to enhance the reliability of optic flow estimation was to perform a *multi–scale* computation. The multi–scale approach proved to be very powerful. In matching methods, it greatly reduces the dimension of the search space. In filtering based methods, it increases the range of measurable displacement magnitudes, and relaxes the need for an a priori tuned frequency or scale parameter.

This work was motivated by the observation that wavelets are a very well designed tool for optic flow measurement. Because of their multiscale structure, and because large scale filterings can be performed efficiently with the fast wavelet transform, they are a very natural tool to measure optic flow: the projection of the optic flow equation onto the wavelets yields a very fast variant of massive filtering.

3.1 Projected differential optic flow estimation

An image sequence is a real function $I(t; x_1, x_2)$ of three variables t, x_1 and x_2 that we suppose continuous in this first section. We use the concise notations $\boldsymbol{x}$ for (x_1, x_2) and $\boldsymbol{x}(t)$ for $(x_1(t), x_2(t))$. The standard mathematical model used to find the optic flow is based on

a *constant brightness assumption*: a real point $[X_1(t)\ \ X_2(t)\ \ X_3(t)]$ in our scene is projected onto the image plane to an image point $(x_1(t), x_2(t))$

$$\left[\ X_1(t) \quad X_2(t) \quad X_3(t)\ \right] \mapsto (x_1(t), x_2(t)) \qquad \text{at time } t$$

The optic flow at time t and location $\boldsymbol{x}(t)$ is defined as the velocity of the image point

$$\boldsymbol{v} = (v_1, v_2) = \left(\frac{dx_1}{dt}, \frac{dx_2}{dt}\right)$$

The brightness constancy assumption consists in saying that the brightness of $(x_1(t), x_2(t))$ at time t is time independent, hence

$$I(t; \boldsymbol{x}(t)) = I_0$$

The optic flow $\boldsymbol{v}$ is therefore constrained by the following equation

$$\frac{\partial I}{\partial t} + \boldsymbol{\nabla} I \cdot \frac{d\boldsymbol{x}}{dt} = 0$$

or

$$\boldsymbol{v} \cdot \boldsymbol{\nabla} I + \frac{\partial I}{\partial t} = 0 \tag{OF}$$

We also define a variant of (OF) that takes possible illumination changes into account. This is done with a Lambertian surface aspect model.

$$I(t; x_1, x_2) = R(t; x_1, x_2) \times L(t; x_1, x_2)$$

In this formula, R is the reflectance, i.e. a picture sequence fulfilling the brightness constancy assumption and therefore (OF). L is the illumination factor that accounts for brightness changes. For a single light source at finite distance, L is a product

$$L(t; x_1, x_2) = \frac{L_0}{d^2} \cos i$$

where i is the angle of incidence of the source light falling on the object, and d is the distance between the source and the object. Changes in illumination are caused by relative moves between the light source and the object. We suppose that these changes have slow variations in space. This consists in assuming that the spatial derivatives $\partial L/\partial x$ and $\partial L/\partial y$ are negligible.

By differentiation, we get

$$\begin{aligned} \frac{dI}{dt} &= \frac{dR}{dt} L + R \frac{\partial L}{\partial t} \\ &= 0 + R \frac{\partial \log L}{\partial t} L \end{aligned}$$

and a modified optic flow equation taking illumination changes into account:

$$\frac{\partial I}{\partial x_1} v_1 + \frac{\partial I}{\partial x_2} v_2 + \frac{\partial I}{\partial t} = \frac{L'}{L} I \tag{OFL}$$

The parameters we can estimate from this constraint are the optic flow $\boldsymbol{v} = (v_1, v_2)$ and the time derivative of $\log L$: L'/L. Note that L cannot be measured more accurately than up to a multiplicative constant, since a pattern $\alpha \times P$ and an illumination $\alpha^{-1} \times L$ give the same sequence I and thus the same constraints. This is reflected in the fact that we only measure the time derivative of $\log L$ which is not affected by a multiplication of L with a constant factor.

Equations (OF) or (OFL) cannot be solved pointwise, because on each location and each time, it consists in solving a single scalar equation for two or more scalar unknowns. This is the *aperture problem*. As long as no a priori information is available on the nature of the underlying motion, the only equation to solve for the optic flow is (OF) or (OFL). The problem of optic flow measurement has infinitely many solutions and is by essence ill-posed.

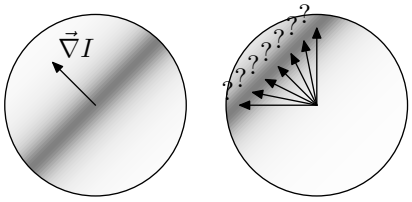


Figure 3.1: Illustration of the aperture problem. A smooth picture can be locally considered as a translation invariant picture. Thus only the projection onto $\boldsymbol{\nabla} I$ of the displacement between $t = 0$ and $t = 1$ can be estimated, while the orthogonal component cannot.

The only way to reduce the number of solutions is to do an additional assumption. This rule applies in all flow measurement techniques and to our knowledge, no method makes an exception to it.

As a first example, Horn and Schunck [HS81] changed this non invertible linear system into a well conditioned one by writing it as a functional minimization problem.

Finding the minimum of

$$M[\boldsymbol{v}] = \iint \left(\boldsymbol{v} \cdot \boldsymbol{\nabla} I + \frac{\partial I}{\partial t} \right)^2 dx_1 dx_2$$

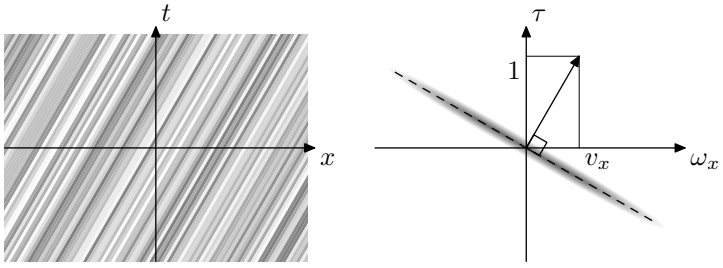
is equivalent to solving (OF). Horn and Schunck add to $M[\boldsymbol{v}]$ a smoothness functional of the form

$$S[\boldsymbol{v}] = \lambda \iint \|\Delta \boldsymbol{v}\|^2 \, dx_1 dx_2$$

where λ is a positive parameter used to tune the trade-off between smoothness and the accordance of $\boldsymbol{v}$ with (OF). The total functional $M[\boldsymbol{v}] + S[\boldsymbol{v}]$ is positive definite and has a unique minimum. As such however, this method cannot be used to estimate wide ranges of displacements.

Methods based on spatiotemporal filtering (see Fig. 3.2) with velocity tuned filters [BRRO94, CLF95, FJ90, GD94, Hee88] also assume that the flow is constant over the

support of their filters. Block matching methods [Ana89, BYX83] rely on the assumption that the motion is constant over small windows of the picture. We make no exception to this rule and have to do a similar assumption to extract the optic flow.



Sequence in (x,t) coordinates Sequence spectrum in (ω_x,τ)

Figure 3.2: Illustration of the principle underlying spatiotemporal filtering based methods. On the left, a picture sequence $I(t;x)$ of uniform velocity is displayed. The spatiotemporal Fourier transform of such a sequence is located on a hyperplane whose tilt indicates the picture motion. The normal vector has coordinates $(v_x,1)$ where v_x is the optic flow. Identifying the tilt of this plane allows to estimate the optic flow.

In this work, our way around aperture is the following: we define some measure functions $(\psi^n)_{n=1\dots N}$ of $L_2(\mathbb{R}^2)$ all centered around $(0,0)$ and with different frequency contents. We compute the inner product of either (OF) or (OFL) against translated functions $\psi^n(\boldsymbol{x}-\boldsymbol{u})$ to obtain N different equations. In the simpler case of (OF), this yields

$$\iint \left(\frac{\partial I}{\partial x_1}v_1(\boldsymbol{x})+\frac{\partial I}{\partial x_2}v_2(\boldsymbol{x})+\frac{\partial I}{\partial t}\right)\psi^n(\boldsymbol{x}-\boldsymbol{u})dx_1dx_2=0 \qquad \forall n=1\dots N \tag{3.1}$$

Using the notation $\langle f,g\rangle=\iint f(\boldsymbol{x})\overline{g(\boldsymbol{x})}dx_1dx_2$, and $\psi^n_{\boldsymbol{u}}(\boldsymbol{x})=\psi^n(\boldsymbol{x}-\boldsymbol{u})$, this can also be written as

$$\left\langle\frac{\partial I}{\partial x_1}v_1,\psi^n_{\boldsymbol{u}}\right\rangle+\left\langle\frac{\partial I}{\partial x_2}v_2,\psi^n_{\boldsymbol{u}}\right\rangle+\left\langle\frac{\partial I}{\partial t},\psi^n_{\boldsymbol{u}}\right\rangle=0 \qquad \forall n=1\dots N \tag{3.2}$$

If we now do the following assumption (or approximation) that $v_1(\boldsymbol{x})$ and $v_2(\boldsymbol{x})$ are constant over the supports of $\psi^n_{\boldsymbol{u}}$, i.e.

$$\begin{aligned} v_1(\boldsymbol{x})&=v_1(\boldsymbol{u}) \qquad \forall \boldsymbol{x}\in \text{support}\,\psi^n_{\boldsymbol{u}},\ \forall n\\ v_2(\boldsymbol{x})&=v_2(\boldsymbol{u}) \qquad \forall \boldsymbol{x}\in \text{support}\,\psi^n_{\boldsymbol{u}},\ \forall n \end{aligned} \tag{$\mathcal{A}$}$$

Equation (3.2) then becomes

$$\left\langle\frac{\partial I}{\partial x_1},\psi^n_{\boldsymbol{u}}\right\rangle v_1(\boldsymbol{u})+\left\langle\frac{\partial I}{\partial x_2},\psi^n_{\boldsymbol{u}}\right\rangle v_2(\boldsymbol{u})+\frac{\partial}{\partial t}\langle I,\psi^n_{\boldsymbol{u}}\rangle=0 \qquad \forall n=1\dots N \tag{3.3}$$

and after an integration by parts

$$\left\langle I, \frac{\partial \psi^n_{\boldsymbol{u}}}{\partial x_1} \right\rangle v_1(\boldsymbol{u}) + \left\langle I, \frac{\partial \psi^n_{\boldsymbol{u}}}{\partial x_2} \right\rangle v_2(\boldsymbol{u}) = \frac{\partial}{\partial t} \left\langle I, \psi^n_{\boldsymbol{u}} \right\rangle \qquad \forall n = 1 \dots N \tag{3.4}$$

We have a *projected system* of N (typically 3 or 4) equations with unknowns $v_1(\boldsymbol{u})$ and $v_2(\boldsymbol{u})$.

$$\left\{ \begin{array}{ccc} \left\langle I, \frac{\partial \psi^1_{\boldsymbol{u}}}{\partial x_1} \right\rangle v_1(\boldsymbol{u}) + \left\langle I, \frac{\partial \psi^1_{\boldsymbol{u}}}{\partial x_2} \right\rangle v_2(\boldsymbol{u}) & = & \frac{\partial}{\partial t} \left\langle I, \psi^1_{\boldsymbol{u}} \right\rangle \\ \vdots & \vdots & \vdots \\ \left\langle I, \frac{\partial \psi^N_{\boldsymbol{u}}}{\partial x_1} \right\rangle v_1(\boldsymbol{u}) + \left\langle I, \frac{\partial \psi^N_{\boldsymbol{u}}}{\partial x_2} \right\rangle v_2(\boldsymbol{u}) & = & \frac{\partial}{\partial t} \left\langle I, \psi^N_{\boldsymbol{u}} \right\rangle \end{array} \right. \tag{$\mathcal{S}$}$$

that has to be compared to the single equation (OF) for $\boldsymbol{x} = \boldsymbol{u}$: now we have found a way around aperture. The price we had to pay to achieve this was the above assumption that the optic flow is constant over the support of the $\psi^n_{\boldsymbol{u}}$. Under some assumptions on the smoothness of the optic flow, we can prove that this estimation method is consistent, i.e. that the extracted flow field converges to the real one as the support size of the measure functions goes to zero. This theorem is stated in Sec. 4.1.2 and proved in appendix.

The motivation for such an approach is multiple. It looks very appealing because we extract with an explicit formula and an inexpensive wavelet measure a direct estimate of the optic flow. As compared to window or multiscale matching methods, it also allows to obtain subpixel displacement accuracy. It is worth emphasizing that the systems we solve are of very limited size, with either 2 or 3 unknowns (3 in the case of illumination change measurement). This is very different from the approach of Horn and Schunck who solve a single system whose unknown is the whole optic flow vector map of dimension $2N$ if the frames have N pixels.

A similar approach has been proposed by Weber and Malik in 1995 [WM95]. They use real valued functions of several scales to filter the OF equation, which makes their approach computationally more expensive. Simoncelli *et al.* [Sim98] proposed a Bayesian estimation resolution method of multiscale differential constraints, and Magarey and Kingsbury [MK98] an approach based on the minimization of subband squared image differences. The former uses a real steerable pyramid as a set of filters, and the latter use analytic wavelets built with filters sampled from Gabor functions. Our approach is very close in spirit to these approaches but differs in several aspects detailed later in this paper, including filter design (Sec. 4.3), measure of illumination, integer motion warping (Sec. 4.1.4) as well as the alias free measurement range extension (Sec. 4.2.2).

Note that compared to some spatiotemporal filtering based methods, this approach supposes that there is a unique motion vector to be extracted. This assumption may be invalidated where several superimposed components of the picture are moving with different velocities, or around occlusions. Spatiotemporal filtering can allow to measure such superimposed velocities as shown if Fig. 3.3. The price to pay for these additional features is twofold: an additional assumption on the time dependence of the optic flow, and increased computational complexity. This issue is for example discussed in [FJ90], and in [Sim92].

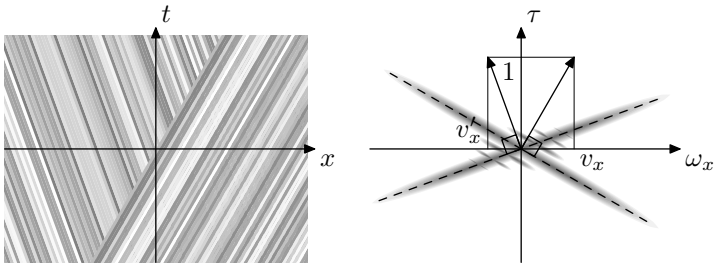


Figure 3.3: This illustrates how a spatiotemporal filtering analysis of an occlusion can several velocity components. On the left, we have displayed in x and t a sequence where a component moving to the right is hiding the component moving to the left. In the right figure, we see the corresponding spectrum which is roughly supported on the union of two hyperplanes. The hyperplane corresponding to the occluded region is somewhat blurred, but both motions can be distinguished.

In our approach however, we build such small local linear systems that are overdetermined: the number of equations is larger than the number of unknowns. This provides us with a mean to check whether our optic flow model is valid over the considered area. If the equations are not compatible, and the system has no solution, this means that our model is not valid (either because the optic flow is varying to fast with respect to the location, or because we are measuring the flow over an occlusion where it is discontinuous). We detail in Sec. 4.1.3 how we deal with such overdetermined systems.

These considerations about the above assumption argue for the use of measure functions ψ of smallest possible support size. However, as we will see in the next section, time aliasing considerations provide an opposite constraint on the support size of the measure functions, which are therefore chosen as the result of a trade-off.

3.2 Time aliasing and measure function scale

In practice, our picture sequence is sampled in time and thus known only for integer values of t, and the right–hand side coefficients in system ($\mathcal{S}$)

$$\frac{\partial}{\partial t}\langle I, \psi^n \rangle$$

must be estimated with a finite difference approximation of the picture time derivative $\partial I/\partial t$ in time, like $\partial I/\partial t \simeq I(t+1) - I(t)$.

We now compare the relative errors for different scalings $\psi^n_{\boldsymbol{us}}(x) = s^{-1}\psi^n(\frac{\boldsymbol{x}-\boldsymbol{u}}{s})$ of the function ψ^n. We will see that the error of such estimations is high if the displacement (v_1, v_2) between two pictures $I(t)$ and $I(t+1)$ is large with respect to the scale s of the measure function $\psi^n_{\boldsymbol{us}}$.

Let us suppose that for a given s, the picture is translating uniformly over the support of our functions $\psi^n_{\boldsymbol{u}s}$ for all n, i.e.

$$I(t;\boldsymbol{x}) = I(\boldsymbol{x} - t\boldsymbol{v})$$

The simplest time derivative approximation is a two step finite difference

$$\frac{\partial I(t)}{\partial t} \simeq I(t+1) - I(t)$$

In this work however, we use a higher order estimate, and measure the optic flow at each $t + 1/2$, between two successive pictures, on the basis of the following approximation:

$$\frac{\partial I(t+1/2)}{\partial t} \simeq I(t+1) - I(t) \tag{3.5}$$

Now we also need to compute coefficients of the left hand side of system ($\mathcal{S}$) at $t + 1/2$:

$$\left\langle \frac{\partial}{\partial x_i}\psi^n_{\boldsymbol{u}s}, I(t+1/2) \right\rangle$$

because we only know them at integer times. This is done with the following estimation:

$$I(t+1/2) \simeq \frac{I(t) + I(t+1)}{2} \tag{3.6}$$

At a given scale s, these approximations lead respectively to the following coefficient approximations:

$$\begin{aligned} \left\langle \frac{\partial I(t+1/2)}{\partial t}, \psi^n_{\boldsymbol{u}s} \right\rangle &\simeq \langle I(t+1) - I(t), \psi^n_{\boldsymbol{u}s} \rangle \\ \langle I(t+1/2), \psi^n_{\boldsymbol{u}s} \rangle &\simeq \left\langle \frac{I(t+1) + I(t)}{2}, \psi^n_{\boldsymbol{u}s} \right\rangle \end{aligned}$$

which can be rewritten after variable changes and integrations by parts

$$\langle I(t+\tfrac{1}{2}), \boldsymbol{v}\cdot\boldsymbol{\nabla}\psi^n_{\boldsymbol{u}s} \rangle \simeq \left\langle I(t+\tfrac{1}{2}), \psi^n_{\boldsymbol{u}s}(\boldsymbol{x}+\boldsymbol{v}/2) - \psi^n_{\boldsymbol{u}s}(\boldsymbol{x}-\boldsymbol{v}/2) \right\rangle \tag{3.7a}$$

$$\langle I(t+\tfrac{1}{2}), \psi^n_{\boldsymbol{u}s} \rangle \simeq \left\langle I(t+\tfrac{1}{2}), \frac{\psi^n_{\boldsymbol{u}s}(\boldsymbol{x}+\boldsymbol{v}/2) + \psi^n_{\boldsymbol{u}s}(\boldsymbol{x}-\boldsymbol{v}/2))}{2} \right\rangle \tag{3.7b}$$

Each approximation (3.7a) and (3.7b) is the approximation of a linear functional of $L_2(\mathbb{R})$ with another one. We take as a design rule that the following approximations be true:

$$\boldsymbol{v}\cdot\boldsymbol{\nabla}\psi^n_{\boldsymbol{u}s} \simeq \psi^n_{\boldsymbol{u}s}(\boldsymbol{x}+\boldsymbol{v}/2) - \psi^n_{\boldsymbol{u}s}(\boldsymbol{x}-\boldsymbol{v}/2) \tag{3.8a}$$

$$\psi^n_{\boldsymbol{u}s} \simeq \frac{\psi^n_{\boldsymbol{u}s}(\boldsymbol{x}+\boldsymbol{v}/2) + \psi^n_{\boldsymbol{u}s}(\boldsymbol{x}-\boldsymbol{v}/2))}{2} \tag{3.8b}$$

With a Taylor expansion of ψ, we can prove that there exists some constant M such that the sum of the relative errors of (3.8a) and (3.8b) is less than $M \times (|\boldsymbol{v}|/s)^2$. This sum has

been numerically estimated for the wavelets we use later in this algorithm (filter in (6.1b)), and if the constraint

$$\boxed{|\boldsymbol{v}| \leq 0.42 \times s} \tag{3.9}$$

is verified, approximations (3.8a) and (3.8b) are valid up to a 15% error. We will see in Sec. 4.2.2 that this range can be shifted, which is necessary for the purpose of large motion compensation, described in Sec. 4.1.4.

Note that this phenomenon always arises in some form in optic flow computation and has been pointed out by many authors [Jäh93]. Also note that this very problem motivates a multiscale approach to optic flow computation in e.g. [Ana89, BRRO94, SF95, WM95, MPM96]. As an example, we see how it shows up in spatiotemporal methods in Fig. 3.4, and in a more general setting in Fig. 3.5.

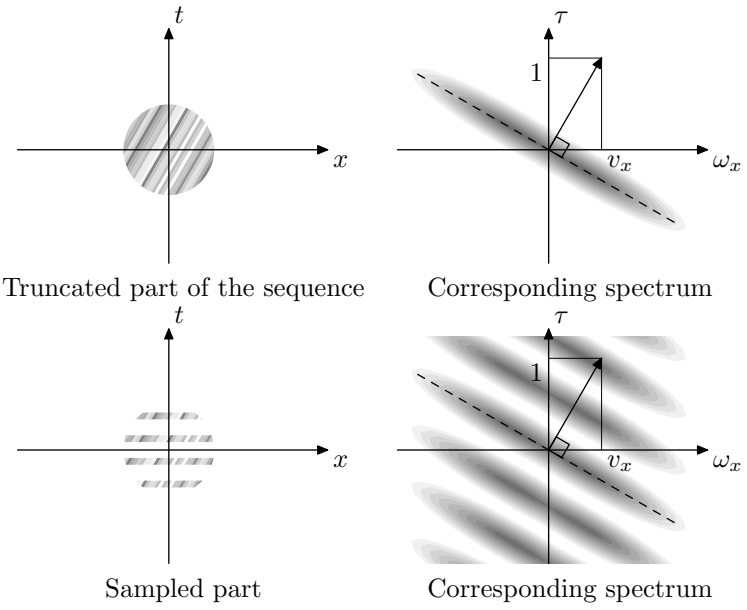


Figure 3.4: These graphs illustrate how time aliasing affects the measure of optic flow in spatiotemporal filtering based methods. The upper-left figure represents a picture sequence of uniform velocity (of Fig. 3.2) truncated in space and time. This results in a blurred spectrum (upper right). If this sequence is now sampled in time, the corresponding spectrum is periodized in time, so that the resulting spectrum is a comb of tilted parallel planes. If the velocity increases, the tilt of these planes increases. The aliasing limit occurs when the planes get too close one to another to be correctly separated.

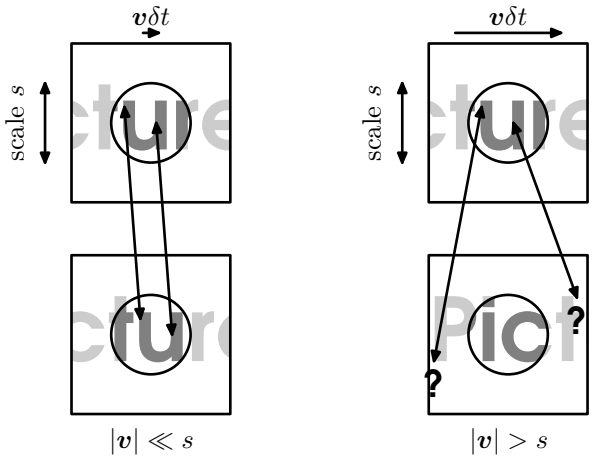


Figure 3.5: This illustrates the aliasing problem in the most general case. Considering only portions of a picture of size s does not allow to measure displacements $\boldsymbol{v}$ of amount $|\boldsymbol{v}|$ larger than s, because then the two parts of the picture considered at times $t = 0$ and $t = 1$ are not related, so no correspondence can be found (right figure).

So if we make the approximations (3.5) and (3.6), the system ($\mathcal{S}$) discretized in time is

$$\left\{ \begin{array}{lcl} \displaystyle\sum_{\ell=1,2} \left\langle \frac{I(t+1)+I(t)}{2}, \frac{\partial \psi^1_{\boldsymbol{u}s}}{\partial x_\ell} \right\rangle v_\ell & = & \left\langle I(t+1) - I(t), \psi^1_{\boldsymbol{u}s} \right\rangle \\ \vdots & \vdots & \vdots \\ \displaystyle\sum_{\ell=1,2} \left\langle \frac{I(t+1)+I(t)}{2}, \frac{\partial \psi^N_{\boldsymbol{u}s}}{\partial x_\ell} \right\rangle v_\ell & = & \left\langle I(t+1) - I(t), \psi^N_{\boldsymbol{u}s} \right\rangle \end{array} \right. \qquad (\mathcal{DS})$$

where v_1 and v_2 are supposed to be measured at location $\boldsymbol{u}$ and time $t + 1/2$.

Remark

Note that the approximation of the picture derivative with a finite difference in (3.5) has to be considered in a weak sense, since the picture may happen to be not differentiable. However, we only use inner products of the form $\langle \nabla I, \psi \rangle$ or $\langle I, \psi \rangle$. In such inner products, the Taylor approximation we write formally on the image function I in (3.5) is transferred with an integration by parts and variable changes on the wavelet ψ in (3.7a) and (3.8a), and since the wavelets we use are several times differentiable, such a Taylor approximation makes again sense.

Bibliography

[AB85] E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of vision. *J. Opt. Soc. Amer.*, A2:284–299, 1985.

[Ana89] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2:283–310, 1989.

[BA83] P. Burt and E. Adelson. The Laplacian pyramid as a compact image code. *IEEE. Trans. Communications*, 31:532–540, 1983.

[BFB94] J. Barron, D. Fleet and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[BRRO94] T. Burns, S. Rogers, D. Ruck and M. Oxley. Discrete, spatiotemporal, wavelet multiresolution analysis method for computing optical flow. *Optical Engineering*, 33(7):2236–2247, 1994.

[BYX83] P. Burt, C. Yen and X. Xu. Multiresolution flow–through motion analysis. In *Proc. Conference Computer Vision and Pattern Recognition*, pages 246–252. Washington, 1983.

[CLF95] C. Clifford, K. Langley and D. Fleet. Centre-frequency adaptive IIR temporal filters for phase-based image velocity estimation. *Image Processing and its Applications*, 4(6):173–177, 1995.

[FJ90] D. Fleet and A. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5:77–104, 1990.

[GD94] M. Gökstorp and P.-E. Danielsson. Velocity tuned generalized Sobel operators for multiresolution computation of optical flow. In *International Conference on Image Processing*, pages 765–769. 1994.

[Hee88] D. Heeger. Optical flow using spatiotemporal filters. *International Journal for Computer Vision*, 1:279–302, 1988.

[HS80] B. Horn and B. Schunck. Determining optical flow. Technical Report A.I. Memo 572, Massachusetts Institute of Technology, 1980.

[HS81] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–204, 1981.

[Jäh93] B. Jähne. Spatio-temporal image processing. In *Lecture Notes in Computer Science*, volume 751. 1993.

[MK98] J. Magarey and N. Kingsbury. Motion estimation using a complex–valued wavelet transform. *IEEE Trans. on Signal Processing*, 46(4):1069–1084, April 1998.

[MPM96] É. Mémin, P. Pérez and D. Machecourt. Dense estimation and object-oriented segmentation of the optical flow with robust techniques. Technical Report 2836, Rapport de Recherches INRIA, France, 1996.

[SF95] E. Simoncelli and W. Freeman. The steerable pyramid: a flexible architecture for multi-scale derivative computation. In *2nd Annual IEEE International Conference on Image Processing, Washington DC*. 1995.

[Sim92] E. P. Simoncelli. Distributed representation of image velocity. Technical Report 202, MIT Media Laboratory, 1992.

[Sim98] E. P. Simoncelli. Bayesian multi–scale differential optical flow. In H. Jähne and Geissler, editors, *Handbook of computer vision and applications*. Academic Press, 1998.

[WM95] J. Weber and J. Malik. Robust computation of optical flow in a multi-scale differential framework. *International Journal of Computer Vision*, 14(1):5–19, 1995.

Chapter 4

Description of our method

4.1 Wavelets as a natural multiscale analysis tool

Since the choice of the function scaling s cannot be done a priori, we use a coarse to fine refinement scheme. This suggests naturally to use a wavelet basis to do such measures, since they have the multiscale and translation invariant structure the optic flow estimation requires.

We start from a set of mother wavelets $(\psi^n)_{n=1\dots N}$ in $L_2(\mathbb{R}^2)$. We then define a discrete wavelet family $(\psi^n_{j\boldsymbol{k}})_{n=1\dots N, j\in\mathbb{Z}, \boldsymbol{k}\in\mathbb{Z}^2}$ by

$$\psi^n_{j\boldsymbol{k}}(\boldsymbol{x}) = 2^j \psi^n(2^j\boldsymbol{x} - \boldsymbol{k})$$

where j is a resolution index, $\boldsymbol{k} = (k_1, k_2)$ a 2-dimensional translation index, and $\boldsymbol{x}$ a 2–dimensional variable $\boldsymbol{x} = (x_1, x_2)$.

> EXAMPLE— In image processing, a set of three real-valued mother wavelets is commonly used. These wavelets are built as tensor products of a scaling function $\phi \in L_2(\mathbb{R})$ and a wavelet $\psi \in L_2(\mathbb{R})$:
>
> $$\psi^1(\boldsymbol{x}) = \psi(x_1)\phi(x_2) \quad (4.1a)$$
> $$\psi^2(\boldsymbol{x}) = \phi(x_1)\psi(x_2) \quad (4.1b)$$
> $$\psi^3(\boldsymbol{x}) = \psi(x_1)\psi(x_2) \quad (4.1c)$$
>
> A typical spectrum for these mother wavelets is displayed in Fig. 4.1.

Note that a wavelet $\psi^n_{j\boldsymbol{k}}$ is located around $(2^{-j}k_1, 2^{-j}k_2)$, and spreads over a domain of size proportional to 2^{-j}.

For any pair $(j, \boldsymbol{k})$, we have N scaled and translated wavelets $\psi^n_{j\boldsymbol{k}}$, $n = 1 \dots N$. We can do the same computations as we already did in equations (3.1)–(3.4). For any $(j, \boldsymbol{k})$, we thus have a *local projected system* of N equations which constrains an average value of the optic

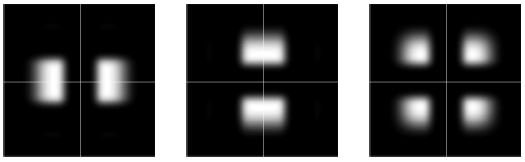


Figure 4.1: Spectrum of standard 2D real wavelets

flow around $2^{-j}\boldsymbol{k}$, and with characteristic scale 2^{-j}

$$\left\{\begin{array}{ccccc} \left\langle I, \dfrac{\partial \psi^1_{j\boldsymbol{k}}}{\partial x_1}\right\rangle v_1 + \left\langle I, \dfrac{\partial \psi^1_{j\boldsymbol{k}}}{\partial x_2}\right\rangle v_2 & = & \left\langle \dfrac{\partial I}{\partial t}, \psi^1_{j\boldsymbol{k}}\right\rangle \\ \vdots & & \vdots & & \vdots \\ \left\langle I, \dfrac{\partial \psi^N_{j\boldsymbol{k}}}{\partial x_1}\right\rangle v_1 + \left\langle I, \dfrac{\partial \psi^N_{j\boldsymbol{k}}}{\partial x_2}\right\rangle v_2 & = & \left\langle \dfrac{\partial I}{\partial t}, \psi^N_{j\boldsymbol{k}}\right\rangle \end{array}\right. \qquad (\mathcal{S}_{j\boldsymbol{k}})$$

In this system, I and $\frac{\partial I}{\partial t}$ are supposed to be estimated at times $t+1/2$ with formulae (3.5) and (3.6) and the system can thus be written in more detail like ($\mathcal{DS}$). Note that as is explained in the previous section and especially (3.9), the flow estimation made by this system can only be valid if the displacement $\boldsymbol{v}$ is small w.r.t 2^{-j}.

The gain in using wavelets arising from a multiresolution analysis is that all coefficients of this system can be computed with dyadic filtering and subsampling schemes. Indeed,

- on the left hand side of ($\mathcal{S}_{j\boldsymbol{k}}$), the coefficients are wavelet derivative coefficients of the picture function I, that can be computed with derivative multiresolution analysis (see Chap. A).
- on the right hand side, the coefficients are standard wavelet coefficients of the picture time derivative. They are estimated with finite differences in time of standard wavelet coefficients of the picture.

4.1.1 Wavelet bases or frames

The wavelet framework provides an easy way to ensure that the wavelet measures we use provide an overcomplete information on the image sequence. This happens whenever our wavelet family is a basis or a *frame*, or more generally a redundant family.

Redundancy (or at least completeness) is required to guarantee that we do not miss any information by switching from the optic flow equation (OF) to the system of all possible equations (3.1) for all wavelet scales, orientations and translations. Nothing guarantees however that we cannot oversee a motion when we isolate a small subset of equations making up a local system ($\mathcal{S}_{j\boldsymbol{k}}$) and trying to solve this subsystem individually. In the most general case, as long as the pattern of the picture region we are considering is rich enough of features, we can reasonably expect such systems to be well defined.

This expectation may not be fulfilled in very special cases. If for example the moving pattern has very different frequency contents in x_1 and x_2 directions, like a pattern

$$I(x, y) = \sin x_1 + \sin 10x_2 \tag{4.2}$$

the motion is not directly measurable with a single system ($\mathcal{S}_{j\boldsymbol{k}}$) at any scale. Wavelets at a given scale j_1 have a given spatial frequency bandwidth, and may only "see" the $\sin x_1$ component of the pattern, i.e.

$$\langle \psi^n_{j_1\boldsymbol{k}}, I\rangle = \langle \psi^n_{j_1\boldsymbol{k}}, \sin x_1\rangle$$

This component of the pattern being translation invariant, the corresponding local system is then underdetermined.

At another scale j_2, the wavelets again only "see" the second component, i.e.

$$\langle \psi^n_{j_2\boldsymbol{k}}, I\rangle = \langle \psi^n_{j_2\boldsymbol{k}}, \sin 10x_2\rangle$$

thus again a translation invariant pattern.

In such a case, it may happen that no local system at any scale provides a way to find out the motion, although the whole set of equations with all scales together provides enough information to measure the displacement of such a pattern. To circumvent such a difficulty, we combine least square constraints of different scales, as is detailed in Sec. 4.1.4.

4.1.2 Convergence results

We can state two convergence results to support our approach. The first result motivates the approximation done by assuming in Sec. 3.1 that the optic flow is constant over the support of the wavelets $\psi^n_{j\boldsymbol{k}}$. We write the system $\mathcal{S}_{j\boldsymbol{k}}$ in short

$$M_{j\boldsymbol{k}}\boldsymbol{v} = Y_{j\boldsymbol{k}}\,.$$

We assume that the picture sequence $I(t;\boldsymbol{x})$ is the result of a smooth transformation

$$I(t;\boldsymbol{x} + \boldsymbol{\delta}(\boldsymbol{x}, t)) = I(0;\boldsymbol{x}) \qquad \text{for all } \boldsymbol{x},\ t \tag{4.3}$$

where $\boldsymbol{\delta}$ is m times continuously differentiable in $(\boldsymbol{x}, t)$ and notice that

$$\boldsymbol{v}(\boldsymbol{x}) = \frac{\partial \boldsymbol{\delta}(\boldsymbol{x}, t)}{\partial t}\,.$$

A first theorem states that we can control the error caused by optic flow variation in space.

Theorem 4.1

Assume $\boldsymbol{x} \mapsto I(0;\boldsymbol{x})$ is locally L_2 everywhere and α–Lipschitz at some point $\boldsymbol{x}_0$. Also assume that $\boldsymbol{x} \mapsto I(0;\boldsymbol{x})$ is not of any higher Lipschitz regularity at $\boldsymbol{x}_0$. Assume that $(\boldsymbol{x}, t) \mapsto \boldsymbol{\delta}(\boldsymbol{x}, t)$ is $\lfloor\alpha\rfloor + 3$ times continuously differentiable. Then

1. *The true velocity* $\boldsymbol{v}(\boldsymbol{x}_0, 0)$ *in* $\boldsymbol{x}_0$ *at time* $t = 0$ *fulfills the following system of equations*

$$\left\{ \begin{array}{rcl} \left\langle I, \dfrac{\partial \psi^1_{j\boldsymbol{k}}}{\partial x_1} \right\rangle v_1(\boldsymbol{x}_0, 0) + \left\langle I, \dfrac{\partial \psi^1_{j\boldsymbol{k}}}{\partial x_2} \right\rangle v_2(\boldsymbol{x}_0, 0) + e^1_{j\boldsymbol{k}} & = & \left\langle \left.\dfrac{\partial I}{\partial t}\right|_{t=0}, \psi^1_{j\boldsymbol{k}} \right\rangle \\ \vdots & \vdots & \vdots \\ \left\langle I, \dfrac{\partial \psi^N_{j\boldsymbol{k}}}{\partial x_1} \right\rangle v_1(\boldsymbol{x}_0, 0) + \left\langle I, \dfrac{\partial \psi^N_{j\boldsymbol{k}}}{\partial x_2} \right\rangle v_2(\boldsymbol{x}_0, 0) + e^N_{j\boldsymbol{k}} & = & \left\langle \left.\dfrac{\partial I}{\partial t}\right|_{t=0}, \psi^N_{j\boldsymbol{k}} \right\rangle \end{array} \right. \tag{4.4}$$

   written in short

$$M_{j\boldsymbol{k}} \boldsymbol{v}(\boldsymbol{x}_0, 0) + E_{j\boldsymbol{k}} = Y_{j\boldsymbol{k}} \,. \tag{4.5}$$

   There exist sequences $j_p \in \mathbb{Z}$, $j_p \to +\infty$, $\boldsymbol{k}_p \in \mathbb{Z}^2$ *such that*

$$2^{-j_p} \boldsymbol{k}_p \to \boldsymbol{x}_0 \qquad\qquad \text{as } p \to +\infty$$

   and the error term $E_{j_p \boldsymbol{k}_p}$ *is negligible with respect to the norm of the system matrix* $M_{j_p \boldsymbol{k}_p}$ *on this sequence*

$$\frac{\|E_{j_p \boldsymbol{k}_p}\|}{\|M_{j_p \boldsymbol{k}_p}\|} \to 0 \qquad\qquad \text{as } p \to +\infty \tag{4.6}$$

2. *Then, if there exists an increasing integer sequence* $q \mapsto p(q)$ *such that the condition number of the square matrices* $M_{j_{p(q)} \boldsymbol{k}_{p(q)}}$ *is bounded, then the solutions* $\boldsymbol{v}_q$ *of the systems*

$$M_{j_{p(q)} \boldsymbol{k}_{p(q)}} \boldsymbol{v}_q = Y_{j_{p(q)} \boldsymbol{k}_{p(q)}} \tag{4.7}$$

   converge to the true velocity $\boldsymbol{v}(\boldsymbol{x}_0)$ *as* $q \to +\infty$.

3. *If the matrices* $M_{j_{p(q)} \boldsymbol{k}_{p(q)}}$ *are not square, and if their condition number with respect to left inversion* $\sqrt{\operatorname{cond} M^T M}$ *are bounded, then the same result holds with* $\boldsymbol{v}_q$ *defined as the least square solution of* (4.7) *which is*

$$\boldsymbol{v}_q = \left({M_{j_{p(q)} \boldsymbol{k}_{p(q)}}}^T M_{j_{p(q)} \boldsymbol{k}_{p(q)}} \right)^{-1} {M_{j_{p(q)} \boldsymbol{k}_{p(q)}}}^T Y_{j_{p(q)} \boldsymbol{k}_{p(q)}}$$

   so that again

$$\boldsymbol{v}_q \to \boldsymbol{v}(\boldsymbol{x}_0, 0) \qquad\qquad \text{as } q \to +\infty$$

A proof for this theorem is given in 5.2.1. Note that it only shows that the error induced by the assumption made in Sec. 3.1 on the flow uniformity in space allows to make an estimation of the true flow which is asymptotically correct. It does not handle estimation error caused by time aliasing. A second theorem also proved in appendix handles the problem of time aliasing.

Theorem 4.2

We make the same assumptions on the image displacement $\boldsymbol{\delta}$ and on the image smoothness at time 0 as in Th. 4.1. If for some $\theta > 0$, t_j is a sequence of times such that $t_j = \mathcal{O}(2^{-j(1+\theta)})$ as $j \to +\infty$, then

1. *the true velocity $\boldsymbol{v}(\boldsymbol{x}_0, 0)$ in $\boldsymbol{x}_0$ at time $t = 0$ fulfills the following system of equations*

$$\left\{ \begin{array}{ccc} \displaystyle\sum_{\ell=1,2} \left\langle I, \frac{\partial \psi^1_{j\boldsymbol{k}}}{\partial x_\ell} \right\rangle v_\ell(\boldsymbol{x}_0, 0) + e'^1_{j\boldsymbol{k}} & = & \left\langle \frac{I(t_j) - I(0)}{t_j}, \psi^1_{j\boldsymbol{k}} \right\rangle \\ \vdots & \vdots & \vdots \\ \displaystyle\sum_{\ell=1,2} \left\langle I, \frac{\partial \psi^N_{j\boldsymbol{k}}}{\partial x_\ell} \right\rangle v_\ell(\boldsymbol{x}_0, 0) + e'^N_{j\boldsymbol{k}} & = & \left\langle \frac{I(t_j) - I(0)}{t_j}, \psi^N_{j\boldsymbol{k}} \right\rangle \end{array} \right. \tag{4.8}$$

   written in short

$$M_{j\boldsymbol{k}} \boldsymbol{v}(\boldsymbol{x}_0, 0) + E'_{j\boldsymbol{k}} = Y'_{j\boldsymbol{k}} \,. \tag{4.9}$$

   There exist sequences $j_p \in \mathbb{Z}$, $j_p \to +\infty$, $\boldsymbol{k}_p \in \mathbb{Z}^2$ such that

$$2^{-j_p} \boldsymbol{k}_p \to \boldsymbol{x}_0 \qquad\qquad \text{as } p \to +\infty$$

   and the error term $E'_{j_p \boldsymbol{k}_p}$ is negligible with respect to the norm of the system matrix $M_{j_p \boldsymbol{k}_p}$ on this sequence:

$$\frac{\|E'_{j_p \boldsymbol{k}_p}\|}{\|M_{j_p \boldsymbol{k}_p}\|} \to 0 \qquad\qquad \text{as } p \to +\infty \tag{4.10}$$

2. *If for some increasing integer sequence $q \mapsto p(q)$, the condition number of the matrices $M_{j_{p(q)} \boldsymbol{k}_{p(q)}}$ is bounded, then the solution $\boldsymbol{v}'_q$ of the systems*

$$M_{j_{p(q)} \boldsymbol{k}_{p(q)}} \boldsymbol{v}'_q = Y'_{j_{p(q)} \boldsymbol{k}_{p(q)}} \tag{4.11}$$

   converge to the true velocity $\boldsymbol{v}(\boldsymbol{x}_0)$ as $q \to +\infty$. Again, this holds when the matrices M are square or when they have more rows than columns, in which case the condition number has to be taken with respect to left inversion.

Remarks

- *The fact that the time step t_j has to be negligible with respect to the grid step 2^{-j} reflects the problem of time aliasing: the inter frame displacement (which is proportional to the time step) has to be negligible with respect to the grid step.*
- *While the smoothness assumptions needed to prove both theorems are rather strong for the displacement mapping $(\boldsymbol{x}, t) \mapsto \boldsymbol{x} + \boldsymbol{\delta}(\boldsymbol{x}, t)$, the assumption on the image itself is fairly weak. We only assume that $\boldsymbol{x} \mapsto I(0; \boldsymbol{x})$ is α–Lipschitz on $\boldsymbol{x}_0$, and is locally L_2 anywhere else.*

4.1.3 Solving overdetermined systems

The linear systems we build are overdetermined: there are more equations than unknowns. We write them as

$$MV = Y \tag{4.12}$$

where M is a complex valued matrix of size $n \times m$, V a vector of optic flow components of size $m \times 1$ and Y a complex valued vector of size $n \times 1$. The integer m is either 2 or 3, depending on whether we also measure illumination ($V = [v_1\, v_2\, L'/L]^T$) or not ($V = [v_1\, v_2]^T$).

First, we build the corresponding least square system. Note that since the matrices M and Y are complex valued, and we are looking for a real valued flow vector V, the system can be written as a real system:

$$\begin{bmatrix} \operatorname{Re} M \\ \operatorname{Im} M \end{bmatrix} V = \begin{bmatrix} \operatorname{Re} Y \\ \operatorname{Im} Y \end{bmatrix}$$

The system is overdetermined, because usually $2n$ is much larger than m. The corresponding least square system is then

$$((\operatorname{Re} M)^* \operatorname{Re} M + (\operatorname{Im} M)^* \operatorname{Im} M)V = ((\operatorname{Re} M)^* \operatorname{Re} Y + (\operatorname{Im} M)^* \operatorname{Im} Y)$$

which can be written in the more concise form

$$\operatorname{Re}(M^* M)V = \operatorname{Re}(M^* Y) \tag{4.13}$$

We can distinguish basically three possible cases for such a system.

1. If this least square system is not of full rank, the system cannot be solved because of aperture.
2. If the least square system can be solved and has a solution V_0, there still are two possible cases.
   (a) Either the solution is also solution of the original system (4.12), i.e. $MV_0 = Y$, from which we conclude that the flow is uniform over the given area.
   (b) Or the solution does not satisfy the original system. This means that our model of optic flow is not valid over the considered area because the flow is varying too fast, or is even discontinuous.

In practice, we find out in which case we are by comparing the condition number of $\operatorname{Re}(M^* M)$ with a first threshold (to decide whether the system is underdetermined or not), and then the fitting error $\|MV_0 - Y\|$ with a second threshold. The fitting error indicates how true our assumption is on the optic flow constancy. It is high if we are looking at a flow discontinuity or if the flow is varying rapidly in space. If we consider using it to do optic flow segmentation, we have to discriminate between both cases, and a richer local flow model is necessary. Instead of a flow model according to which the flow is supposed to be locally constant, we can use a model of a locally linear flow that is motivated by a basic 3D solid motion model by Kanatani [Kan89] (see Sec. 6.4).

4.1.4 Coarse to fine refinement scheme

Our local systems are located at grids of several resolutions 2^j. The corresponding discrete grids $\{2^{-j}(k_1, k_2)\}$ are displayed in figure 4.2. At each node of each of these grids, we have a local system ($\mathcal{S}_{j\boldsymbol{k}}$). The issue we address now is the strategy to combine information

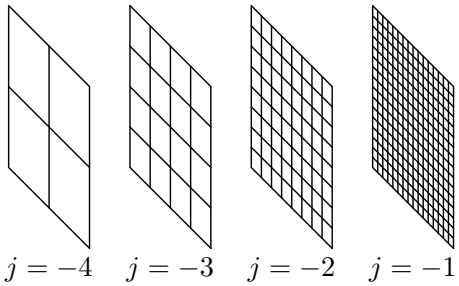


Figure 4.2: Grids at several resolutions

coming from several resolutions to construct a "flat" flow map at the finest resolution. To sum up, measures obtained at different resolutions have different characteristics:

— Finer scale measures provide (1) a finer knowledge of the space dependence of the optic flow and (2) rely on a looser assumption ($\mathcal{A}$) on the flow field uniformity.

— However, coarse scale measurement are less subject to time aliasing as indicated in (3.9).

In this approach, we follow the path already opened by several authors in various flow measurement methods [BYX83, Ana89, WM95, Sim98, MK98] to combine informations from several scales in the following way: large displacements are measured first and at large scales. Then, the resulting motion is compensated so that the residual motion is smaller and stays within the alias free range of the finer scale subsystems.

Assume a motion estimation at scale $j-1$ has predicted a motion of $\tilde{\boldsymbol{v}}$. The predicted motion $\tilde{\boldsymbol{v}}$ can for example be uniquely decomposed as

$$\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}^N + \tilde{\boldsymbol{v}}^r$$

where

$$\begin{cases} \tilde{\boldsymbol{v}}^N \in 2^{-j}\mathbb{Z}^2 \\ \tilde{\boldsymbol{v}}^r \in 2^{-j}[-0.5, 0.5)^2 \end{cases}$$

The motion estimation refinement at scale j then consists in finding the motion $\boldsymbol{v}$ as $\tilde{\boldsymbol{v}}^N + \boldsymbol{v}^r$ where we expect the new residual motion $\boldsymbol{v}^r$ to be not much larger that the older one $\tilde{\boldsymbol{v}}^r$. The residual motion $\boldsymbol{v}^r = (v_1^r, v_2^r)$ is estimated as a solution of the system of the

following N equations

$$\frac{1}{2}\left(\left\langle I(t+1), \frac{\partial \psi^n_{j,\boldsymbol{k}+2^j\tilde{\boldsymbol{v}}^N}}{\partial x_1}\right\rangle + \left\langle I(t), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_1}\right\rangle\right) v_1^r + \frac{1}{2}\left(\left\langle I(t+1), \frac{\partial \psi^n_{j,\boldsymbol{k}+2^j\tilde{\boldsymbol{v}}^N}}{\partial x_2}\right\rangle + \left\langle I(t), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_1}\right\rangle\right) v_2^r = \left\langle I(t+1), \psi^n_{j,\boldsymbol{k}+2^j\tilde{\boldsymbol{v}}^N}\right\rangle - \left\langle I(t), \psi^n_{j\boldsymbol{k}}\right\rangle \quad (4.14)$$

for $n = 1 \dots N$.

These equations can be compared with the same in $\boldsymbol{v} = (v_1, v_2)$ when no large motion compensation is done (which consists in setting $\tilde{\boldsymbol{v}}^N = 0$), and which are

$$\frac{1}{2}\left(\left\langle I(t+1), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_1}\right\rangle + \left\langle I(t), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_1}\right\rangle\right) v_1 + \frac{1}{2}\left(\left\langle I(t+1), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_2}\right\rangle + \left\langle I(t), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_1}\right\rangle\right) v_2 = \left\langle I(t+1), \psi^n_{j\boldsymbol{k}}\right\rangle - \left\langle I(t), \psi^n_{j\boldsymbol{k}}\right\rangle \quad (4.15)$$

This coarse to fine refinement scheme is initialized with a coarsest scale measurement in which the predicted motion $\tilde{\boldsymbol{v}}$ is set to 0.

Note that large motion compensation is done only partially: in integer multiples of the current grid step so that no interpolation computation has to be done. In this, our approach differs from that of Simoncelli [Sim98] or Magarey and Kingsbury [MK98].

The behavior of a local system at a given resolution j and location $2^{-j}(k_1, k_2)$ gives several kinds of information on our measurements:

1. If the system has a unique solution (v_1, v_2), we get a straightforward estimation of the optic flow at $2^{-j}\boldsymbol{k}$. This happens if our underlying model of uniform flow is valid.

2. If the system has no solution, it means that our assumption ($\mathcal{A}$) is incorrect. In such a case, finer scale measurements are necessary, since they rely on a looser uniformity assumption ($\mathcal{A}$). As an estimate of the current scale, we keep a least square estimate of the velocity that is required for large motion compensation.

   > If for example our measure region overlaps two regions where the flow is different, we have to split our measure region in subregions, to perform again this measure on each of these subregions, where hopefully the optic flow is more uniform.

3. If on the contrary, the system has infinitely many solutions, this means that the pattern at the location we are looking at is too poor to tell us how it is moving. The aperture problem remains.

   > A typical case is a locally translation invariant pattern, because then it is impossible to measure its motion along its translation invariance axis.

In practice, we make no difference between the cases 1 and 2. Moreover, as motivated in Sec. 4.1.1, we combine constraints on the flow coming from several resolutions. If a system we get at scale j (of the form (4.13)) is written $A_j V = B_j$ where A_j is symmetric positive, we combine this system with that of the preceding scale at the very same location $A_{j-1}V = B_{j-1}$, and solve at scale j the system

$$(A_j + \rho A_{j-1})\, V = B_j + \rho B_{j-1} \tag{4.16}$$

Such a combination makes sense because solving a system $AV = B$ where A is symmetric positive is equivalent to minimizing the quadratic functional $V \mapsto V^T AV - 2B^T V$, and solving the above combined system is thus equivalent to minimizing a positive combination of such functionals. In practice, the coefficient ρ is chosen to be relatively small, so that the LS constraints coming from scale $j-1$ influence the flow estimation at scale j only when constraints obtained at scale j fail to determine properly the flow.

As a safeguard against errors induced by time aliasing, we add a test that is performed in cases 1 or 2. We reject measures (v_1^r, v_2^r) (or (v_1, v_2) if no large motion compensation is done) that are above the time aliasing threshold, i.e. for which $|(v_1, v_2)| > \alpha \times 2^{-j}$.

Remarks

- *Usually, the system we pick up from the coarser scale $j-1$: $A_{j-1}V = B_{j-1}$ is expressed in terms of another residual motion, so that an affine change of variables has to be done prior to the combination of LS constraints described in (4.16).*
- *The above decomposition of a global motion as a sum of an integer step motion plus a residual motion makes sense if a residual motion can always be measured within a range which is strictly larger than some $2^{-j}[-0.5, 0.5)^2$. For this purpose, the aliasing limit given in (3.9) is not sufficient. This aliasing range can be extended by oversampling the wavelet transform, and/or shifted by using* frequency shifting*, as described in Sec. 4.2.2.*

4.2 Analytic wavelets

In this section, we explain how using real valued wavelets in the framework described in our paper does not allow to perform stable measurements of the optic flow, and motivate the use of analytic wavelets. We will see that this allows us to gather a higher number of constraints on the optic flow, and that the corresponding measures can be also made less sensitive to time aliasing. Let us first briefly recall some notations on analytic functions and the Hilbert transform.

The Hilbert transform H maps $L_2(\mathbb{R}, \mathbb{C})$ onto $L_2(\mathbb{R}, \mathbb{C})$ and can be defined in the time or the frequency domain, by either of formulae

$$\widehat{Hf}(\omega) = \hat{f}(\omega) \times \mathrm{sign}(\omega) \tag{4.17a}$$

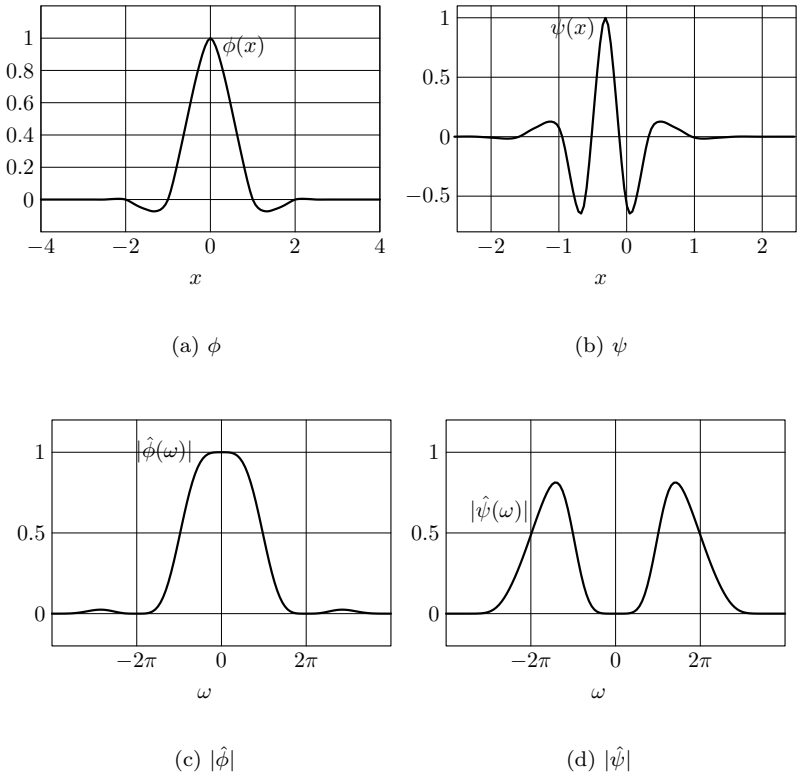
$$Hf = f * \left(\frac{2}{it}\right) \tag{4.17b}$$

A function f is analytic if its negative frequency content is 0, which is the same as saying that it is invariant under the Hilbert transform. A real valued signal is never analytic, unless it is zero. The analytic transform of a real valued signal f is $f^+ = f + Hf$. Note that the real signal can be recovered from the analytic one by taking its real part: $f = \mathrm{Re}\, f^+$.

4.2.1 Estimation stability

Standard monovariate real valued wavelets are displayed in figure (4.3–a-d). If we use wavelets ψ^1, ψ^2 and ψ^3 as defined in (4.1a–4.1c) to compute the optic flow, the coefficients of the system of equations ($\mathcal{S}$) are real valued and highly oscillating in space.

For these reasons, they happen to vanish very often, or to be below the noise level, which makes our estimation of the flow unstable.



(a) ϕ (b) ψ

(c) $|\hat{\phi}|$ (d) $|\hat{\psi}|$

Figure 4.3: ϕ, ψ and their Fourier transforms

To demonstrate this in more detail, we go back to the one–dimensional case. The velocity

estimation can be written as

$$v(k/2^j) \simeq \frac{\frac{\partial}{\partial t}\int I(t;x)\psi_{jk}(x)dx}{\int I(t;x)\psi'_{jk}(x)dx} \tag{4.18}$$

where ψ is a classical real-valued wavelet, as displayed in figure (4.3–a,c). We denote by ψ^+ its analytic counterpart. The denumerator in (4.18) can be written as

$$D(k/2^j) = \left[I(t) * \psi_{j0}\right](k/2^j) \tag{4.19}$$

or as

$$D(k/2^j) = \mathrm{Re}\left[I(t) * \psi^+_{j0}\right](k/2^j) \tag{4.20}$$

Since ψ^+ is analytic, $I(t)*\psi^{+\prime}_{j0}$ also is. We decompose this analytic signal into an analytic amplitude and instantaneous frequency, as described in [Mal97], chap. 4:

$$\left[I(t) * \psi^{+\prime}_{j0}\right](k/2^j) = A(k/2^j)e^{i\theta(k/2^j)} \tag{4.21}$$

In this decomposition, A is a positive function called analytic amplitude, θ is the phase function and its derivative θ' is the instantaneous frequency.

In the case where we use the real valued wavelet ψ in the estimation (4.18), the denumerator can be written as the real part of the above term, and its absolute value is therefore

$$|D(k/2^j)| = A(k/2^j)|\cos\theta(k/2^j)| \tag{4.22}$$

The cosine part usually oscillates at a frequency of about 2^j, which makes it very likely to be close to 0 for several integers k. Fig. 4.4–a shows that sampled values of $|D(k/2^j)|$ can be close to 0: estimations of ratios with such a denumerator are as reliable as playing *Russian roulette*.

If we now use an analytic wavelet ψ^+ instead of the real one, the formula giving v becomes

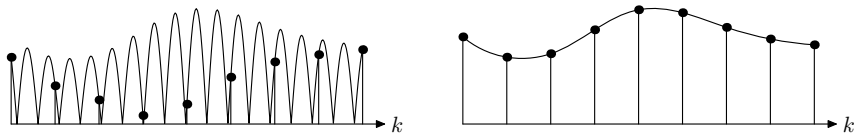
$$v(k/2^j) \simeq \frac{\frac{\partial}{\partial t}\int I(t;x)\psi^+_{jk}(x)dx}{\int I(t;x)\psi^{+\prime}_{jk}(x)dx} \tag{4.23}$$

The denumerator is thus now

$$D(k/2^j) = A(k/2^j)e^{i\theta(k/2^j)} \tag{4.24}$$

and its absolute value simply $A(k/2^j)$. This new denumerator is now much less likely to vanish (see Fig. 4.4–b).

Whereas the above argument is formally valid only in 1D, the experience showed that the same problem appears in 2D (and it is a strong understatement to say that the first numerical experiments with real valued classical wavelet (4.1a–4.1c) were not very encouraging).



(a) real case (b) analytic case

Figure 4.4: Sampling of $|D(k/2^j)|$ in the real and analytic cases

Two–dimension analytic wavelets This suggests us to replace our three real wavelets ψ^1, ψ^2 and ψ^3 as defined in equations (4.1a–4.1c), with the four following wavelets

$$\Psi^1(\boldsymbol{x}) = \psi^+(x_1)\phi(x_2) \tag{4.25a}$$
$$\Psi^2(\boldsymbol{x}) = \phi(x_1)\psi^+(x_2) \tag{4.25b}$$
$$\Psi^3(\boldsymbol{x}) = \psi^+(x_1)\psi^+(x_2) \tag{4.25c}$$
$$\Psi^4(\boldsymbol{x}) = \psi^+(x_1)\psi^-(x_2) \tag{4.25d}$$

where $\psi^-(x) = \overline{\psi^+(x)}$.

Theorem 4.3

If $(\psi^n_{j\boldsymbol{k}})_{n=1\ldots3,j\in\mathbb{Z},\boldsymbol{k}\in\mathbb{Z}^2}$ is a basis of $L_2(\mathbb{R}^2,\mathbb{R})$, then $(\Psi^n_{j\boldsymbol{k}})_{n=1..4,j\in\mathbb{Z},\boldsymbol{k}\in\mathbb{Z}^2}$ is a redundant family of $L_2(\mathbb{R}^2,\mathbb{R})$: there exists some positive bound M such that

$$\sum_{j\boldsymbol{k}n} |\langle f, \Psi^n_{j\boldsymbol{k}}\rangle|^2 \geq M_1\|f\|_2^2 \qquad \text{for any } f \text{ in } L_2(\mathbb{R}^2) \tag{4.26}$$

We also state a converse stability result: the energy of the wavelet coefficients of some function f is bounded by some multiple of the L_2 norm or f:

$$\sum_{j\boldsymbol{k}n} |\langle f, \Psi^n_{j\boldsymbol{k}}\rangle|^2 \leq M_2\|f\|_2^2 \qquad \text{for any } f \text{ in } L_2(\mathbb{R}^2) \tag{4.27}$$

A proof for inequality (4.26) is given below, while the more technical proof for inequality (4.27) is given in appendix in section 5.1.

Proof. Noticing that

$$\langle\psi^1, I\rangle = \mathrm{Re}\langle\Psi^1, I\rangle$$
$$\langle\psi^2, I\rangle = \mathrm{Re}\langle\Psi^2, I\rangle$$
$$\langle\psi^3, I\rangle = \mathrm{Re}\langle\Psi^3, I\rangle + \mathrm{Re}\langle\Psi^4, I\rangle$$

we have

$$|\langle\psi^1, I\rangle|^2 + |\langle\psi^2, I\rangle|^2 \leq |\langle\Psi^1, I\rangle|^2 + |\langle\Psi^2, I\rangle|^2$$

and

$$\begin{aligned}|\langle\psi^3, I\rangle|^2 &\leq \left(\langle\Psi^3, I\rangle + \langle\Psi^4, I\rangle\right)^2 \\ &\leq 2\left(\langle\Psi^3, I\rangle^2 + \langle\Psi^4, I\rangle^2\right)\end{aligned}$$

Both inequalities imply that

$$\sum_{j\boldsymbol{k}n} |\langle f, \Psi^n_{j\boldsymbol{k}}\rangle|^2 \geq \frac{1}{2}\sum_{j\boldsymbol{k}n}\langle f, \psi^n_{j\boldsymbol{k}}\rangle^2$$

and the result follows from the fact that $(\psi^n_{j\boldsymbol{k}})$ is a basis. □

Remark

Note that the wavelets Ψ^n are complex valued. They are therefore not in $L_2(\mathbb{R}^2, \mathbb{R})$, and we cannot strictly say that the wavelet family makes up a frame of $L_2(\mathbb{R}^2, \mathbb{R})$. A more rigorous way to state this would be either to say that

$$\left\{\mathrm{Re}(\Psi^n_{j\boldsymbol{k}}),\, \mathrm{Im}(\Psi^n_{j\boldsymbol{k}}) : n = 1 \ldots N,\ j \in \mathbb{Z} \text{ and } \boldsymbol{k} \in \mathbb{Z}^2\right\}$$

is a frame of $L_2(\mathbb{R}^2, \mathbb{R})$, or that

$$\left\{\Psi^n_{j\boldsymbol{k}},\, \overline{\Psi^n_{j\boldsymbol{k}}} : n = 1 \ldots N,\ j \in \mathbb{Z} \text{ and } \boldsymbol{k} \in \mathbb{Z}^2\right\}$$

is a frame of $L_2(\mathbb{R}^2, \mathbb{C})$.

This allows us to ensure in some way that this representation does not miss some motion information, as is discussed in section 4.1.1.

Analytic measure functions are also used in spatiotemporal filtering techniques, where velocity tuned filters are analytic [FJ90]. Note, however, that the Hilbert transform is also used to make filters direction selective and not analytic [WAA85, BRRO94].

Using analytic wavelets in such a formulation, we integrate in a synthetic way both energy and phase output of our filters into out systems, and step out of the debate on whether the use of the phase output or the energy output of the filters is best suited for measuring optic flow. With a similar argument, Magarey and Kingsbury also motivate the use of analytic wavelets [MK98].

Psychophysical evidence also supports the use of analytic wavelets. Daugman [Dau88] identified a pair of (real valued) Gabor filters with a $\pi/2$ phase shift between them

$$\begin{aligned} f_1 &= e^{-\|\boldsymbol{x}-\boldsymbol{x}_0\|^2/2\sigma^2}\cos\boldsymbol{\omega}\cdot\boldsymbol{x} \\ f_2 &= e^{-\|\boldsymbol{x}-\boldsymbol{x}_0\|^2/2\sigma^2}\sin\boldsymbol{\omega}\cdot\boldsymbol{x}\end{aligned}$$

Such a pair can equivalently be viewed as a single complex filter

$$f = e^{-\|\boldsymbol{x}-\boldsymbol{x}_0\|^2/2\sigma^2} e^{i\boldsymbol{\omega}\cdot\boldsymbol{x}} \tag{4.28}$$

that now has a non-symmetric spectrum, and is thus an approximation of an analytic transform of f_1 (provided that the absolute value of the vector $\boldsymbol{\omega}$ is larger than the frequency spread $1/\sigma$ of the Gaussian window).

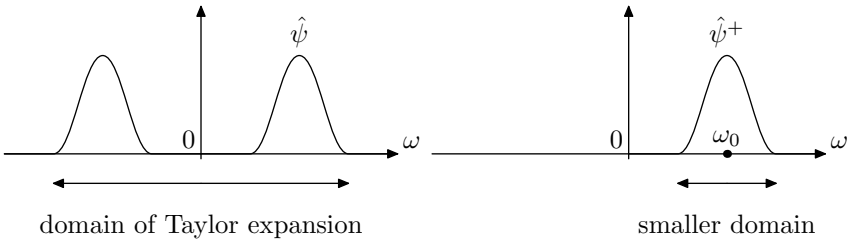
4.2.2 Shifting the alias free measure range with analytic wavelets

Time aliasing can be viewed as an error in estimating the derivative $\boldsymbol{v} \cdot \boldsymbol{\nabla}\psi(\boldsymbol{x})$ with a finite difference $\psi(\boldsymbol{x}) - \psi(\boldsymbol{x} - \boldsymbol{v}\delta t)$. In the Fourier domain, this is the error in approximating a complex exponential introduced by a finite translation in space $e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}}$ with its first order Taylor expansion around 0 which is $1 + i\boldsymbol{v} \cdot \boldsymbol{\omega}$. We remind that $\boldsymbol{v}$ is the displacement per frame $\boldsymbol{v} = (v_1, v_2)$ and denote by $\boldsymbol{\omega}$ the 2-D frequency variable:

$$\left.\begin{array}{c} \psi(\boldsymbol{x}) - \psi(\boldsymbol{x} - \boldsymbol{v}\delta t) \\ \simeq \\ \boldsymbol{v} \cdot \boldsymbol{\nabla}\psi\delta t \end{array}\right\} \Leftrightarrow \left\{\begin{array}{c} \hat{\psi}(\boldsymbol{\omega})(1 - e^{-i\boldsymbol{\omega}\cdot\boldsymbol{v}\delta t}) \\ \simeq \\ \hat{\psi}(\boldsymbol{\omega}) \times (i\boldsymbol{\omega} \cdot \boldsymbol{v}\delta t) \end{array}\right.$$

We show in this section that we can reduce the negative impact of time aliasing by doing the same Taylor approximation of $(1 - e^{i\boldsymbol{\omega}\cdot\boldsymbol{v}\delta t})$, but around a shifted origin in frequency.

Real valued wavelets ψ have a symmetric spectrum, i.e. $\hat{\psi}(\boldsymbol{\omega}) = \overline{\hat{\psi}(-\boldsymbol{\omega})}$. To approximate $\hat{\psi}e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}}$ with a first order Taylor expansion in $\boldsymbol{\omega}$, the most reasonable is to expand $e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}}$ around 0. Analytic wavelets, as described in Eqs. (4.25a–4.25d) have a non symmetric spectrum whose support is not centered on the origin any more. If we do the Taylor expansion of $e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}}$ around the frequency center of these analytic wavelets, the resulting approximation of $e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}}\hat{\psi}(\boldsymbol{\omega})$ is more accurate, because it relies on the validity of a Taylor approximation in $\boldsymbol{\omega}$ over a smaller frequency range (compare Fig. 4.5–a and 4.5–b that illustrate this for monovariate wavelets).



(a) Real wavelets (b) Analytic wavelets

Figure 4.5: Taylor approximation frequency ranges (for 1D wavelets). We notice that on the left hand side, the spectrum of ψ is symmetric, and that $e^{iv\omega}\hat{\psi}$ is most reasonably approximated with a Taylor expansion of $e^{iv\omega}$ in ω around 0. Conversely, on the right side, the spectrum of ψ^+ in not symmetric, and an expansion of $e^{iv\omega}$ around ω_0 yields a more accurate estimation of $e^{iv\omega}\hat{\psi}$.

If we have a look at one of the equations of system ($\mathcal{DS}$) with a discrete wavelet $\psi^n_{j\boldsymbol{k}}$

instead of a continuous one $\psi^n_{\boldsymbol{us}}$, it can be written compactly as

$$\sum_{\ell=1,2} \left\langle \frac{I(t+1)-I(t)}{2}, \frac{\partial \psi^n_{\boldsymbol{jk}}}{\partial x_\ell} \right\rangle v_\ell = \langle I(t+1)-I(t), \psi^n_{\boldsymbol{jk}} \rangle \tag{4.29}$$

Let us see how (4.29) can be inferred with a first order Taylor expansion in the frequency domain around 0. We start from

$$I(t+1/2;\boldsymbol{x}) = I(t+1;\boldsymbol{x}+\boldsymbol{v}/2) \tag{4.30}$$

and

$$I(t+1/2;\boldsymbol{x}) = I(t;\boldsymbol{x}-\boldsymbol{v}/2) \tag{4.31}$$

We infer from (4.31)

$$\begin{aligned}\langle I(t+1/2), \psi^n_{\boldsymbol{jk}} \rangle &= \langle \hat{I}(t), e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}/2} \hat{\psi}^n_{\boldsymbol{jk}} \rangle \\ \langle I(t+1/2), \psi^n_{\boldsymbol{jk}} \rangle &\simeq \left\langle \hat{I}(t), \left(1 + i\frac{\boldsymbol{v}\cdot\boldsymbol{\omega}}{2}\right) \hat{\psi}^n_{\boldsymbol{jk}} \right\rangle\end{aligned}$$

by Taylor approximation

$$\langle I(t+1/2), \psi^n_{\boldsymbol{jk}} \rangle \simeq \langle I(t), \psi^n_{\boldsymbol{jk}} \rangle + \frac{1}{2} \sum_{\ell=1,2} \left\langle I(t), \frac{\partial \psi^n_{\boldsymbol{jk}}}{\partial x_\ell} \right\rangle v_\ell \tag{4.32}$$

From (4.30), we obtain in a similar way

$$\langle I(t+1/2), \psi^n_{\boldsymbol{jk}} \rangle \simeq \langle I(t+1), \psi^n_{\boldsymbol{jk}} \rangle - \frac{1}{2} \sum_{\ell=1,2} \left\langle I(t+1), \frac{\partial \psi^n_{\boldsymbol{jk}}}{\partial x_\ell} \right\rangle v_\ell \tag{4.33}$$

By comparison of (4.33) and (4.32), we finally get (4.29).

If we now do the same computations, but with a shifted Taylor expansion center $\boldsymbol{\omega}^n_{\boldsymbol{jk}}$ (the frequency center of the analytic wavelet $\psi^n_{\boldsymbol{jk}}$), we get with very similar computations

$$\langle I(t+1/2), \psi^n_{\boldsymbol{jk}} \rangle \simeq e^{-i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{\boldsymbol{jk}}/2} \left((1 + i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{\boldsymbol{jk}}/2) \langle I(t), \psi^n_{\boldsymbol{jk}} \rangle + \frac{1}{2} \sum_{\ell=1,2} \langle I(t), \frac{\partial \psi^n_{\boldsymbol{jk}}}{\partial x_\ell} \rangle v_\ell \right)$$

and

$$\langle I(t+1/2), \psi^n_{\boldsymbol{jk}} \rangle \simeq e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{\boldsymbol{jk}}/2} \left((1 - i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{\boldsymbol{jk}}/2) \langle I(t+1), \psi^n_{\boldsymbol{jk}} \rangle - \frac{1}{2} \sum_{\ell=1,2} \langle I(t+1), \frac{\partial \psi^n_{\boldsymbol{jk}}}{\partial x_\ell} \rangle v_\ell \right)$$

that yield by comparison again

$$\sum_{\ell=1,2} \left\langle \frac{e^{-i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}} I(t+1) + e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}} I(t)}{2}, \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_\ell} \right\rangle v_\ell$$
$$= e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}}(1 - i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}) \left\langle I(t+1), \psi^n_{j\boldsymbol{k}} \right\rangle - e^{-i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}}(1 + i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}) \left\langle I(t), \psi^n_{j\boldsymbol{k}} \right\rangle$$

We make a second order approximation by crossing out $e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}}(1 - i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}})$ and its conjugate from this equation, to finally get

$$\sum_{\ell=1,2} \left\langle \frac{e^{-i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}} I(t+1) + e^{i\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}} I(t)}{2}, \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_\ell} \right\rangle v_\ell = \left\langle I(t+1), \psi^n_{j\boldsymbol{k}} \right\rangle - \left\langle I(t), \psi^n_{j\boldsymbol{k}} \right\rangle \qquad (4.34)$$

This formula allows to approximate with a better accuracy the displacement $\boldsymbol{v}$ provided that some estimation of $\boldsymbol{v}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}$ is available. Its use is not straightforward, because it relies on an a priori estimation of the displacement, which seems contradictory. Given a rough estimation $\tilde{\boldsymbol{v}}$ of the velocity obtained with (4.29) or from a coarser scale measurement, a finer estimate can be extracted from the system of equations (4.34) where $\boldsymbol{v}$ has been replaced with $\tilde{\boldsymbol{v}}$ in all complex exponentials:

$$\sum_{\ell=1,2} \left\langle \frac{e^{-i\tilde{\boldsymbol{v}}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}} I(t+1) + e^{i\tilde{\boldsymbol{v}}\cdot\boldsymbol{\omega}^n_{j\boldsymbol{k}}} I(t)}{2}, \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_\ell} \right\rangle v_\ell = \left\langle I(t+1), \psi^n_{j\boldsymbol{k}} \right\rangle - \left\langle I(t), \psi^n_{j\boldsymbol{k}} \right\rangle \qquad (4.35)$$

We call this a measuring range shifting, because instead of measuring velocities with a small aliasing error in a range $|\boldsymbol{v}| \leq \alpha 2^{-j}$, we can now measure them in the shifted range $|\boldsymbol{v} - \tilde{\boldsymbol{v}}| \leq \alpha 2^{-j}$ (for estimates $2^j\tilde{\boldsymbol{v}}$ or reasonable length).

Also note that estimation formula (4.35) without frequency shifting is the same as (4.29) when the a priori estimated $\tilde{\boldsymbol{v}}$ is 0. So using formula (4.29) with a rough approximation $\tilde{\boldsymbol{v}}$ (where e.g. $|\boldsymbol{v} - \tilde{\boldsymbol{v}}|/|\boldsymbol{v}| \leq 30\%$) is still much better than doing no frequency shifting at all.

This method has been tested on a two scale flow measurement where the coarser scale flow estimation is used as an input $\tilde{\boldsymbol{v}}$ in the finer scale flow measurement formula. The result is compared with a $\tilde{\boldsymbol{v}} = 0$ input (i.e. no frequency shift) on figure 4.6.

4.3 Dyadic filter bank wavelets

For computational efficiency, we need wavelets implementable with dyadic filter banks, so that the computation of the system coefficients in $(\mathcal{S}_{j\boldsymbol{k}})$ can be done with a fast wavelet transform. We use separable wavelets $\psi(x_1, x_2) = f(x_1)g(x_2)$, and can therefore limit the scope of our designing to monovariate wavelets. For this, we use all the framework of multiresolution analyses as described by Mallat in [Mal89b, Mal89a, Mal97].

Wavelet coefficients in the one–dimensional case can be computed with a dyadic pyramid filtering and subsampling scheme when the wavelet is an infinite convolution of discrete FIR[1]

[1] finite impulse response

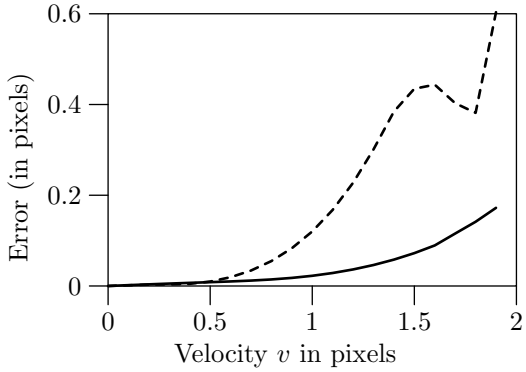


Figure 4.6: Accuracy gain with the frequency shift correction. Compared average errors in flow measurement: with frequency shift (solid) and without (dashed).

filters, which can be written in the Fourier domain as

$$\hat{\psi}(\omega) = \prod_{j=1}^{+\infty} M_j\left(\frac{\omega}{2^j}\right) \tag{4.36}$$

where the M_j's are trigonometric polynomials. For computational efficiency purposes, the functions M_j should be all the same, up to the very first ones.

If we have a wavelet defined with the above filter banks, we describe in App. A a way to design filter banks for the derivative wavelet. In this section, we explain how to design analytic wavelet filter banks.

4.3.1 Analytic dyadic filter bank wavelets

Using a true Hilbert transform to compute analytic wavelet coefficients is not possible in practice because of its computational cost. This would indeed involve functions of infinite support and of very slow decay at infinity. The purpose of this section is to approximate the Hilbert transform ψ^+ of a real wavelet ψ with an *almost analytic* wavelet $\psi^\#$ that can still be implemented with a FIR[1] filter bank.

4.3.2 Design

We want our wavelet $\psi^\#$ to have most of its energy on the positive frequency peak, and we want to keep the relationship $\psi = \mathrm{Re}(\psi^\#)$, the same way as for the true Hilbert transform, $\psi = \mathrm{Re}(\psi^+)$, so that Th. 4.3 remains true when we replace ψ^+ with $\psi^\#$.

We start from any FIR filter pair (m_0, m_1) that can be conjugate mirror filters as used by Adelson and Simoncelli [ASH87] or Mallat [Mal89a] or more generally simply a low- and high-pass pair of filter. We define a wavelet ψ as an infinite convolution of these filters,

which we write in the Fourier domain

$$\hat{\psi}(\omega) = m_1\left(\frac{\omega}{2}\right)\prod_{j=2}^{+\infty} m_0\left(\frac{\omega}{2^j}\right) \quad . \tag{4.37}$$

ψ and its Fourier transform are displayed in (4.3–b,d).

Almost analytic transformation

If now m_2 is a Deslauriers-Dubuc interpolation filter, fulfilling the following relationships:

$$m_2(\omega) + m_2(\omega+\pi) = 2 \qquad \forall \omega \in \mathbb{R}, \text{ and} \tag{4.38}$$

$$m_2(\omega) = m_2(-\omega) \in \mathbb{R} \qquad \forall \omega \in \mathbb{R}, \tag{4.39}$$

then $\hat{\psi}^{\#}(\omega) = \hat{\psi}(\omega) m_2(\omega/4 - \pi/2)$ is a good approximation of $\hat{\psi}^{+}(\omega)$ for two reasons:

- Most of the negative frequency peak of ψ is canceled by a vanishing $m_2(\omega)$. The Fourier transform of ψ and $m_2(\omega)$ are displayed in Fig. 4.7–a. The remaining negative frequency content of $\psi^{\#}$ is not 0, but is less than 2% of $\psi^{\#}$'s total L_2 norm. The Fourier transform of $\psi^{\#}$ is displayed in Fig. 4.7–b, and $\psi^{\#}$ in Fig. 4.7–c.
- We can also retrieve the original wavelet ψ from the almost analytic one $\psi^{\#}$, in the same manner as we can for the truly analytic one. In other words, the same way as $\psi = \mathrm{Re}\,\psi^{+}$, we now have

  $$\psi = \mathrm{Re}\,\psi^{\#}$$

  Indeed,

$$\begin{aligned}\widehat{\left(\psi^{\#} + \overline{\psi^{\#}}\right)}(\omega) &= \widehat{\psi^{\#}}(\omega) + \overline{\widehat{\psi^{\#}}}(-\omega) \\ &= \hat{\psi}(\omega) \times \left(m_2\left(\frac{\omega}{4} - \frac{\pi}{2}\right) + m_2\left(-\frac{\omega}{4} - \frac{\pi}{2}\right)\right) \\ &= 2\hat{\psi}(\omega) \qquad \text{thanks to (4.38–4.39).}\end{aligned}$$

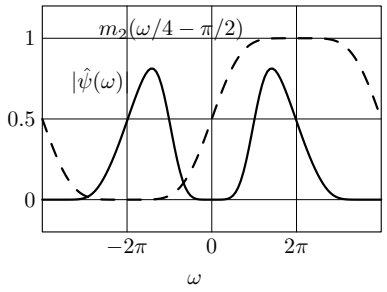
This relationship allows to extend Theorem 4.3 to the case of almost analytic wavelets, so we can state

Theorem 4.4

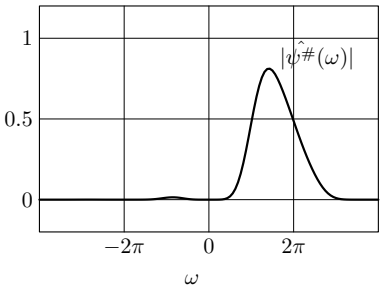
The family of almost analytic bivariate wavelets defined as in (4.25–a-d) by replacing ψ^{+} with $\psi^{\#}$ is a frame.

One of the frame bounds can be obtained exactly the same way as in the proof of Th. 4.3, while the other is proved in 5.1. A similar construction has been suggested by A. Cohen in [Coh92]. His approach is different in that all filtering steps are changed to make the wavelet quasi–analytic: the low–pass filter is replaced with an analytic low–pass filter of infinite support. In our case, a single filtering step is made analytic.

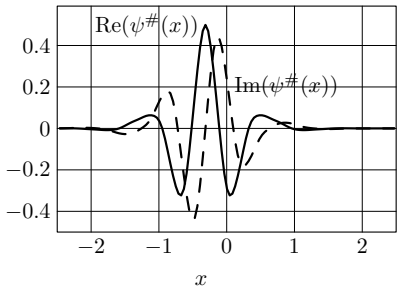
The Fourier transform of the 2D wavelets computed from the 1D analytic wavelets as described in (4.25a–4.25d) now only have a single peak. They are displayed in Fig. 4.8.



(a) $\hat{\psi}$ and $m_2(\frac{\omega}{2} - \frac{\pi}{4})$

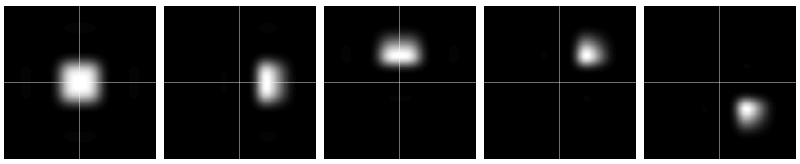


(b) $|\widehat{\psi^{\#}}|$



(c) $\psi^{\#}$

Figure 4.7: Approximation $\psi^{\#}$ of ψ^{+} and its Fourier transform



(a) ψ^0 (b) ψ^1 (c) ψ^2 (d) ψ^3 (e) ψ^4

Figure 4.8: Fourier transform of 2D analytic wavelets

Filtering scheme

Inner products $\int I(x)\psi^{\#}(x)dx$ are computed the same way as $\int I(x)\psi(x)dx$ up to a single additional discrete filtering step. As compared to the product expansion of $\hat{\psi}$

$$\hat{\psi}(\omega) = m_1\left(\frac{\omega}{2}\right) \times m_0\left(\frac{\omega}{4}\right) \times m_0\left(\frac{\omega}{8}\right) \times \dots \ ,$$

the expansion of $\widehat{\psi^{\#}}$ is

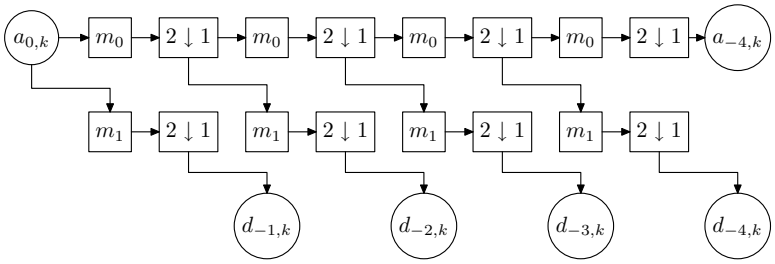
$$\widehat{\psi^{\#}}(\omega) = \hat{\psi}(\omega) m_2\left(\frac{\omega}{4} - \frac{\pi}{2}\right) \tag{4.40}$$

$$= m_1\left(\frac{\omega}{2}\right) \times \left[m_0\left(\frac{\omega}{4}\right) \underbrace{m_2\left(\frac{\omega}{4} - \frac{\pi}{2}\right)}_{\text{additional filter}} \right] \times m_0\left(\frac{\omega}{8}\right) \times m_0\left(\frac{\omega}{16}\right) \times \dots \tag{4.41}$$

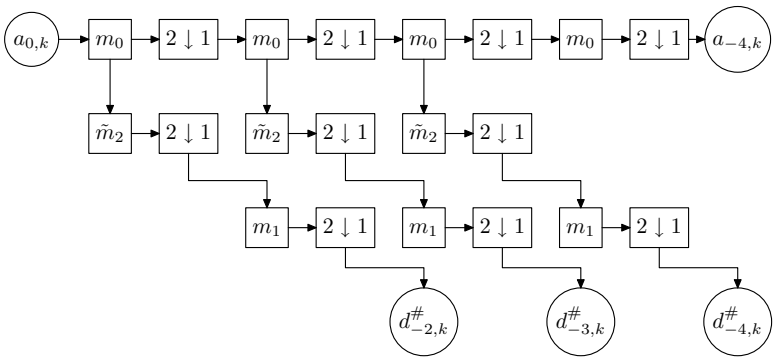
and has an additional discrete filter at the fore-last filtering pass emphasized with a lower bracket. If we write

$$a_{jk}[f] = \langle f, \phi_{jk} \rangle$$
$$d_{jk}[f] = \langle f, \psi_{jk} \rangle$$
$$d^{\#}_{jk}[f] = \langle f, \psi^{\#}_{jk} \rangle$$

the classical filtering scheme to compute coefficients $d_{jk}[f]$ is displayed in Fig. 4.9–a. This can be compared to the analytic patched scheme that is displayed in Fig. 4.9–b. Both figures display filtering and subsampling pyramid schemes using the following conventions: boxes $\boxed{2 \downarrow 1}$ denote dyadic subsampling, and $\boxed{m_0}$ and alike denote filtering steps.



(a) Real



(b) Analytic

Figure 4.9: Filtering schemes in 1D

Bibliography

[Ana89] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2:283–310, 1989.

[ASH87] E. Adelson, E. Simoncelli and R. Hingorani. Orthogonal pyramid transforms for image coding. In *Visual Communications and Image Processing II*, number 845 in SPIE, pages 50–58. 1987.

[BRRO94] T. Burns, S. Rogers, D. Ruck and M. Oxley. Discrete, spatiotemporal, wavelet multiresolution analysis method for computing optical flow. *Optical Engineering*, 33(7):2236–2247, 1994.

[BYX83] P. Burt, C. Yen and X. Xu. Multiresolution flow–through motion analysis. In *Proc. Conference Computer Vision and Pattern Recognition*, pages 246–252. Washington, 1983.

[Coh92] A. Cohen. *Ondelettes et traitement numérique du signal*. Masson, 1992.

[Dau88] J. Daugman. Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression. *IEEE Trans. Acoust., Speech, Signal Processing*, 36(7):1169–1179, 1988.

[FJ90] D. Fleet and A. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5:77–104, 1990.

[Kan89] K. Kanatani. *Group-theoretical methods in image understanding*. Springer Series in Information Sciences. Springer, 1989.

[Mal89a] S. G. Mallat. Multifrequency channel decomposition of images and wavelet models. *IEEE Trans. Acoust., Speech, Signal Processing*, 37(12):2091–2110, December 1989.

[Mal89b] S. G. Mallat. A theory for multiscale signal decomposition: The wavelet representation. *IEEE Trans. on Pattern and Machine Intelligence*, 11(7):674–693, 1989.

[Mal97] S. G. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1997.

[MK98] J. Magarey and N. Kingsbury. Motion estimation using a complex–valued wavelet transform. *IEEE Trans. on Signal Processing*, 46(4):1069–1084, April 1998.

[Sim98] E. P. Simoncelli. Bayesian multi–scale differential optical flow. In H. Jähne and Geissler, editors, *Handbook of computer vision and applications*. Academic Press, 1998.

[WAA85] A. Watson and J. A.J. Ahumada. Model of human visual-motion sensing. *Journal of Optical Society of America*, 2(2):322–342, 1985.

[WM95] J. Weber and J. Malik. Robust computation of optical flow in a multi-scale differential framework. *International Journal of Computer Vision*, 14(1):5–19, 1995.

Chapter 5

Proof of convergence

5.1 Analytic wavelet frame

In these sections, we prove that the wavelets built with the general near–analytic transformation described in Sec. 4.3.1 make up frames.

The frame condition for a given family $\psi^n_{j\boldsymbol{k}}$ is the double inequality

$$M_1\|f\|_2^2 \leq \sum_{n=1}^{N}\sum_{j\in\mathbb{Z}}\sum_{\boldsymbol{k}\in\mathbb{Z}^2} |\langle f, \psi^n_{j\boldsymbol{k}}\rangle|^2 \leq M_2\|f\|_2^2 \tag{5.1}$$

that has to be valid for a given pair of frame bounds M_1 and M_2 and for any function f in L_2. We call the left inequality the *coercion* inequality and the right one the *stability* inequality.

Since our analytic wavelet family has been built in a redundant way, the left inequality is simple. It has been omitted for the one–dimensional case and has been given for 2–dimensional case in Sec. 4.2 for truly analytic wavelets. As explained in Sec. 4.3.1, the proof holds in the same way for our near–analytic wavelets. Note that in this proof, we can relax the assumption that the wavelet system $(\psi^n_{j\boldsymbol{k}})$ from which we start be orthonormal. We only need $(\psi^n_{j\boldsymbol{k}})$ to be an unconditional basis of biorthogonal wavelets.

We now focus on the right hand side inequality, which we prove for our near–analytic wavelets in a quite general case. Let us first precise shortly some notations: we use discrete filters m_ℓ for several indices ℓ. For a given ℓ, m_ℓ indifferently denotes the discrete filter itself (i.e. a discrete sequence of complex values) or its Fourier transform (a 2π–periodic function). The distinction between the two forms is made if necessary by using distinct brackets for the discrete sequence and the Fourier form. The value of the Fourier form is written $m_\ell(\omega)$ for some $\omega \in [0, 2\pi]$, while the discrete sequence is written $m_\ell[k]$, $k \in \mathbb{Z}$. So we have

$$m_\ell[k] = \frac{1}{2\pi}\int_0^{2\pi} m_\ell(\omega)e^{i\omega k}d\omega$$

The support of such a filter is by default the time support, i.e. a subset of $\mathbb{Z}$.

Let us first prove a useful lemma:

Lemma 5.1

Let ψ be a function of $L_2(\mathbb{R}^2)$, with bounded Fourier transform. Assume that $\hat{\psi}(\boldsymbol{\omega})$ decays sufficiently fast as $|\boldsymbol{\omega}| \to \infty$ or $|\boldsymbol{\omega}| \to 0$, namely

$$|\hat{\psi}(\boldsymbol{\omega})| \leq \frac{C}{(1+|\omega_x|)^s(1+|\omega_y|)^s} \qquad \text{for some } s > 1/2 \tag{5.2}$$

$$|\hat{\psi}(\boldsymbol{\omega})| \leq C'|\boldsymbol{\omega}|^\alpha \qquad \text{for some } \alpha > 0 \tag{5.3}$$

Then, there exists a positive and finite bound M_2 such that

$$\sum_{j\in\mathbb{Z}}\sum_{k\in\mathbb{Z}} |\langle \psi_{j\boldsymbol{k}}, f\rangle|^2 \leq M_2\|f\|_2^2 \qquad \text{for any } f \text{ in } L_2(\mathbb{R},\mathbb{C}) \tag{5.4}$$

Proof. We follow the steps of Albert Cohen in [Coh92]. We compute the j–band energy of the expansion:

$$\begin{aligned}
\sigma_j(f) &= \sum_{\boldsymbol{k}\in\mathbb{Z}^2} |\langle f, \psi_{j\boldsymbol{k}}\rangle|^2 \\
&= \frac{1}{4\pi^2}\sum_{\boldsymbol{k}\in\mathbb{Z}^2} |\langle \hat{f}, \hat{\psi}_{j\boldsymbol{k}}\rangle|^2 && \text{(Parseval)} \\
&= \frac{2^{-j}}{4\pi^2}\sum_{\boldsymbol{k}\in\mathbb{Z}^2} \left| \iint_{\mathbb{R}^2} \hat{f}(\boldsymbol{\omega})\hat{\psi}(2^{-j}\boldsymbol{\omega}) e^{-i2^{-j}\boldsymbol{k}\cdot\boldsymbol{\omega}} d\boldsymbol{\omega}\right|^2 \\
&= \frac{2^{j}}{4\pi^2}\sum_{\boldsymbol{k}\in\mathbb{Z}^2} \left| \iint_{\mathbb{R}^2} \hat{f}(2^{j}\boldsymbol{\omega})\hat{\psi}(\boldsymbol{\omega}) e^{-i\boldsymbol{k}\cdot\boldsymbol{\omega}} d\boldsymbol{\omega}\right|^2 && \text{(Variable change)}
\end{aligned}$$

We apply again the Parseval formula to the periodized function of $L_2([0,2\pi]^2)$

$$F(\boldsymbol{\omega}) = \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \hat{f}\left(2^j(\boldsymbol{\omega}+2k\pi)\right)\hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)$$

to obtain

$$\sigma_j(f) = \frac{2^j}{2\pi}\int_0^{2\pi} \left| \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \hat{f}\left(2^j(\boldsymbol{\omega}+2\boldsymbol{k}\pi)\right)\hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)\right|^2 d\boldsymbol{\omega}$$

Choosing some real ρ in $[0,2)$ such that $\rho s > 1$, we write by triangle inequality

$$\begin{aligned}
\sigma_j(f) &\leq \frac{2^j}{2\pi}\iint_{[0,2\pi]^2} \left| \sum_{\boldsymbol{k}\in\mathbb{Z}^2} |\hat{f}\left(2^j(\boldsymbol{\omega}+2\boldsymbol{k}\pi)\right)| \times |\hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)|\right|^2 d\boldsymbol{\omega} \\
&= \frac{2^j}{2\pi}\iint_{[0,2\pi]^2} \left| \sum_{\boldsymbol{k}\in\mathbb{Z}^2} |\hat{f}\left(2^j(\boldsymbol{\omega}+2\boldsymbol{k}\pi)\right)| \times |\hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)|^{1-\rho/2}|\hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)|^{\rho/2}\right|^2 d\boldsymbol{\omega}
\end{aligned}$$

By applying Cauchy–Schwartz's inequality, we obtain

$$\sigma_j(f) \leq \frac{2^j}{2\pi} \iint_{[0,2\pi]^2} \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \left| \hat{f}\left(2^j(\boldsymbol{\omega}+2\boldsymbol{k}\pi)\right)^2 \hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)^{2-\rho} \right| \times \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \left|\hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)\right|^{\rho} d\boldsymbol{\omega}$$

The decay rate of $\hat{\psi}$ assumed in (5.2) ensures that the sums

$$\sum_{\boldsymbol{k}\in\mathbb{Z}^2} \left|\hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)\right|^{\rho} d\boldsymbol{\omega}$$

are uniformly bounded by some C''. We can now write

$$\begin{aligned}
\sigma_j(f) &\leq C'' \frac{2^j}{2\pi} \iint_{[0,2\pi]^2} \sum_{\boldsymbol{k}\in\mathbb{Z}^2} \left| \hat{f}\left(2^j(\boldsymbol{\omega}+2\boldsymbol{k}\pi)\right)^2 \hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)^{2-\rho} \right| d\boldsymbol{\omega} \\
&= C'' \frac{2^j}{2\pi} \int_{\mathbb{R}} \left| \hat{f}\left(2^j(\boldsymbol{\omega}+2\boldsymbol{k}\pi)\right)^2 \hat{\psi}(\boldsymbol{\omega}+2\boldsymbol{k}\pi)^{2-\rho} \right| d\boldsymbol{\omega} \\
&= \frac{C''}{2\pi} \int_{\mathbb{R}} \left|\hat{f}(\boldsymbol{\omega})\right|^2 \times \left|\hat{\psi}(2^{-j}\boldsymbol{\omega})\right|^{2-\rho}
\end{aligned}$$

Now, both assumptions (5.2) and (5.3) guarantee that the series

$$\sum_{J_1\leq j\leq J_2} |\hat{\psi}(2^{-j}\boldsymbol{\omega})|^{2-\rho}$$

converge for both $J_1 \to -\infty$ and $J_2 \to +\infty$ with geometric rate, and that the resulting sum

$$S(\boldsymbol{\omega}) = \sum_{j\in\mathbb{Z}} |\hat{\psi}(2^{-j}\boldsymbol{\omega})|^{2-\rho}$$

is uniformly bounded over some frequency ring $A \leq |\boldsymbol{\omega}| \leq 2A$ by some C'''. Since $S(2\boldsymbol{\omega}) = S(\boldsymbol{\omega})$, this sum is bounded over the whole space $\mathbb{R}^2$ by the same C'''.

As a result,

$$\sum_{j\in\mathbb{Z}} \sigma_j(f) \leq \frac{C''C'''}{2\pi} \iint_{\mathbb{R}^2} \left|\hat{f}(\boldsymbol{\omega})\right|^2 d\boldsymbol{\omega}$$

$$\text{hence} \sum_{j\in\mathbb{Z}} \sum_{k\in\mathbb{Z}^2} |\langle f, \psi_{j\boldsymbol{k}}\rangle|^2 \leq \frac{C''C'''}{2\pi} \|f\|_2^2$$

which finishes the proof. □

Now, let ϕ and ψ be a scaling function and a wavelet defined through

$$\begin{aligned}
\hat{\phi}(\omega) &= \prod_{k=1}^{+\infty} m_0(\omega/2^k) \\
\hat{\psi}(\omega) &= m_1(\omega/2)\hat{\phi}(\omega/2)
\end{aligned}$$

We assume that m_0 is such that $\hat{\phi}(\omega) = \mathcal{O}(1/|\omega|)$ as $\omega \to \pm\infty$, which is a mild smoothness assumption on ϕ. We also assume that m_1 is a high pass filter of finite support, which implies $m_1(\omega) = \mathcal{O}(\omega)$ as $\omega \to 0$. We also assume that

$$\{\psi_{jk_1}(x)\phi_{jk_2}(y),\ \phi(x)_{jk_1}\psi_{jk_2}(y),\ \psi_{jk_1}(x)\psi_{jk_2}(y) : j, k_1, k_2 \in \mathbb{Z}\}$$

is an unconditional basis of $L_2(\mathbb{R}^2)$. We then define

1. a monovariate almost analytic wavelet $\psi^{\#}$, with

$$\hat{\psi}^{\#}(\omega) = p(\omega)\hat{\psi}(\omega)$$

   where p is either the Fourier transform of the Hilbert function $p(\omega) = 2 \times 1_{(\omega>0)}$ or is defined as $p(\omega) = m_2(\omega/4 - \pi/2)$ where m_2 is the filter described in section 4.3.1;

2. and a set of bivariate analytic wavelets

$$\psi^1(\boldsymbol{x}) = \psi^{\#}(x_1)\phi(x_2) \qquad \psi^2(\boldsymbol{x}) = \phi(x_1)\psi^{\#}(x_2)$$
$$\psi^3(\boldsymbol{x}) = \psi^{\#}(x_1)\psi^{\#}(x_2) \qquad \psi^4(\boldsymbol{x}) = \psi^{\#}(x_1)\overline{\psi^{\#}(x_2)}$$

   and for any $n \geq 5$, $\psi^n(\boldsymbol{x}) = \overline{\psi^{n-4}(\boldsymbol{x})}$.

We can now state

Theorems 4.3,4.4

The almost analytic wavelet family $(\psi^n_{j\boldsymbol{k}})$ is a frame of $L_2(\mathbb{R}^2, \mathbb{C})$.

Proof. The assumptions we made on m_0, m_1 and the boundedness of $\omega \mapsto p(\omega)$ clearly imply that any ψ^n fulfills the hypotheses (5.2) and (5.3) of Lemma 5.1. As a consequence, we can bound

$$\sum_{j\in\mathbb{Z}} \sum_{\boldsymbol{k}\in\mathbb{Z}^2} |\langle f, \psi^n_{j\boldsymbol{k}}\rangle|^2$$

by some multiple of $\|f\|_2^2$ for any n, and thus write the stability inequality

$$\sum_{n=1}^{8} \sum_{j\in\mathbb{Z}} \sum_{\boldsymbol{k}\in\mathbb{Z}^2} |\langle f, \psi^n_{j\boldsymbol{k}}\rangle|^2 \leq M_2\|f\|_2^2$$

The coercion frame inequality has already be proved in Sec. 4.2, Th. 4.3 and the proof is thus complete. □

If our function f is real–valued (which is the case of our images), then this expansion is uselessly redundant, since a coefficient $\langle f, \psi^n_{j\boldsymbol{k}}\rangle$ is the conjugate of $\langle f, \psi^{n+4}_{j\boldsymbol{k}}\rangle$ if $n \leq 4$. This is why we only use the 4 first mother wavelets in our approach.

5.2 Consistency of optic flow system

In this section, we prove that approximations we make in the measurement process are consistent, i.e. that the errors involved get relatively small as the time step goes to zero, and the scale to infinity. The main difficulty is to compare the magnitude of a coefficient we estimate with the error we make in that estimation. It is not possible to prove that the relative error goes to zero, but only that the error has a higher decay rate than the coefficient we estimate. A second difficulty is to find out how the time step and the wavelet scale have to converge respectively to 0 and to $+\infty$ in order to ensure the consistency.

To estimate the magnitude of wavelet coefficients around a given location, we use a theorem of Jaffard relating pointwise Lipschitz regularity and wavelet coefficient decay. We remind that a function f of L_2 is said to be α–Lipschitz at location $\boldsymbol{x}_0$ if and only if there exists a polynomial P such that

$$|f(\boldsymbol{x}) - P(\boldsymbol{x})| = \mathcal{O}(|\boldsymbol{x} - \boldsymbol{x}_0|^\alpha) \qquad \text{as } \boldsymbol{x} \to \boldsymbol{x}_0 \tag{5.5}$$

Note that we can then take a polynomial P of degree at most $\lfloor \alpha \rfloor$. Also note that for any $\alpha' < \alpha$, f is then also α'–Lipschitz. The function f is said to be *exactly* α–Lipschitz at $\boldsymbol{x}_0$ if it is not α'–Lipschitz[1] at $\boldsymbol{x}_0$ for any $\alpha' > \alpha$.

5.2.1 Error caused by the space variations of optic flow

We prove in this section the result stated in Th. 4.1. We recall the assumptions we do on the picture model. The image sequence is locally the result of a smooth displacement $(\boldsymbol{x}, t) \mapsto \boldsymbol{\delta}(\boldsymbol{x}, t)$ of a picture $I(0; \boldsymbol{x})$ at time $t = 0$:

$$\boldsymbol{x} \mapsto \boldsymbol{x} + \boldsymbol{\delta}(\boldsymbol{x}, t) \tag{5.6}$$

We assume that $\boldsymbol{x} \mapsto I(0; \boldsymbol{x})$ is locally L_2, and of pointwise regularity α–Lipschitz at $\boldsymbol{x}_0$, which can be written

$$I(\boldsymbol{x}) = P(\boldsymbol{x}) + \mathcal{O}(|\boldsymbol{x} - \boldsymbol{x}_0|^\alpha) \qquad \text{as } \boldsymbol{x} \to \boldsymbol{x}_0 \tag{5.7}$$

where P is a polynomial of degree less than α. We also assume that $I(0)$ is exactly α–Lipschitz at $\boldsymbol{x}_0$, i.e. it is not α'–Lipschitz for any $\alpha' > \alpha$.

Since $\boldsymbol{\delta}(\boldsymbol{x}, 0) = \boldsymbol{x}$ for any $\boldsymbol{x}$, the mapping $\boldsymbol{x} \mapsto \boldsymbol{x} + \boldsymbol{\delta}(\boldsymbol{x}, t)$ is invertible for time $t = 0$, it is also invertible for a range of times around 0 and at the vicinity of any fixed point $\boldsymbol{x}_0$, by means of the local inversion theorem. We write this inverse

$$\boldsymbol{y} \mapsto \boldsymbol{y} - \boldsymbol{\epsilon}(\boldsymbol{y}, t)$$

We assume the displacement is uniformly $(\alpha + 3)$–Lipschitz in $(\boldsymbol{x}, t)$ (and thus $\lfloor \alpha \rfloor + 3$ times continuously differentiable).

[1] In a more general setting, we could define α as the Lipschitz exponent of f at $\boldsymbol{x}_0$, so that f is α'–Lipschitz for $\alpha' < \alpha$ and not for any $\alpha' > \alpha$, in which case f might or might not fulfill Eq. (5.5) for the critical value $\alpha' = \alpha$. The little more generality being gained at the expense of a longer statement, we will not use it here.

At time $t = 0$, the optic flow is then the time derivative:

$$\boldsymbol{v}(\boldsymbol{x}, 0) = \left.\frac{\partial \boldsymbol{\delta}(\boldsymbol{x}, t)}{\partial t}\right|_{t=0} \tag{5.8}$$

In this framework, the brightness constancy assumption is written

$$I(t; \boldsymbol{x} + \boldsymbol{\delta}(\boldsymbol{x}, t)) = I(0; \boldsymbol{x}) \qquad \forall \boldsymbol{x},\, t \tag{5.9}$$

and by time derivation, we get the differential optic flow constraint

$$\boldsymbol{v}(\boldsymbol{x}, 0) \cdot \boldsymbol{\nabla} I + \frac{\partial I}{\partial t} = \mathbf{0} \tag{5.10}$$

The wavelet frame we use for the measurement is written $(\psi^n_{j\boldsymbol{k}})_{j\boldsymbol{k}n}$ and the wavelets are supposed to have at least $\lfloor \alpha \rfloor$ vanishing moments. We also assume that this frame fulfills the hypotheses of Th. A.3, and that the wavelets are continuously differentiable and compactly supported.

In a first lemma, we build a linear system the real flow fulfills up to an approximation error we estimate.

Lemma 5.2

Let α be a positive real. If $\boldsymbol{x} \mapsto I(0; \boldsymbol{x})$ is exactly α–Lipschitz at $\boldsymbol{x}_0$, then we have for any compactly supported wavelet ψ^n an approximate optic flow constraint:

$$v_1(\boldsymbol{x}_0, 0) \left\langle I(0), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_1} \right\rangle + v_2(\boldsymbol{x}_0, 0) \left\langle I(0), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_2} \right\rangle + r(\psi^n, j, \boldsymbol{k}) = \left\langle \frac{\partial I}{\partial t}(0), \psi^n_{j\boldsymbol{k}} \right\rangle \tag{5.11}$$

where

$$|r(\psi^n, j, \boldsymbol{k})| \leq M 2^{-j(\alpha+1)} \left(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1}\right) \tag{5.12}$$

and for any $\alpha' > \alpha$, there exists some strictly positive real N and sequences j_p, $\boldsymbol{k}_p$ and n_p such that

$$\begin{aligned} j_p &\to +\infty && \text{as } p \to +\infty \\ 2^{-j_p} \boldsymbol{k}_p &\to \boldsymbol{x}_0 && \text{as } p \to +\infty \end{aligned}$$

and

$$\max_{\ell=1,2} \left| \left\langle \frac{\partial I(0)}{\partial x_\ell}, \psi^{n_p}_{j_p \boldsymbol{k}_p} \right\rangle \right| \geq N 2^{-j_p \alpha'} \left(1 + |2^{j_p} \boldsymbol{x}_0 - \boldsymbol{k}_p|^{\alpha'}\right) \tag{5.13}$$

Proof. The proof of (5.12) is based on a Taylor expansion of the velocity $\boldsymbol{v}$ around $\boldsymbol{x}_0$. A first order expansion yields

$$\boldsymbol{v}(\boldsymbol{x}) = \boldsymbol{v}(\boldsymbol{x}_0) + R(\boldsymbol{x})(\boldsymbol{x} - \boldsymbol{x}_0)$$

where $R(\boldsymbol{x})$ is 2×2 matrix and $\boldsymbol{x} \mapsto R(\boldsymbol{x})$ has one level of smoothness less than $\boldsymbol{v}$ has, i.e. is $\lfloor \alpha \rfloor + 1$ times continuously differentiable.

The inner product of (5.10) with the wavelets $\psi^n_{j\boldsymbol{k}}$ gives

$$\iint \boldsymbol{v}(\boldsymbol{x}) \cdot \boldsymbol{\nabla} I(0;\boldsymbol{x}) \overline{\psi^n_{j\boldsymbol{k}}(\boldsymbol{x})} d\boldsymbol{x} + \iint \frac{\partial I(0;\boldsymbol{x})}{\partial t} \overline{\psi^n_{j\boldsymbol{k}}(\boldsymbol{x})} d\boldsymbol{x} = 0$$

From this, we obtain (5.11) if we set

$$r(\psi^n, j, \boldsymbol{k}) = -\iint [R(\boldsymbol{x})(\boldsymbol{x} - \boldsymbol{x}_0)] \cdot \boldsymbol{\nabla} I(0;\boldsymbol{x}) \overline{\psi^n_{j\boldsymbol{k}}(\boldsymbol{x})} d\boldsymbol{x}$$

We do an integration by parts to get

$$r(\psi^n, j, \boldsymbol{k}) = \iint I(0;\boldsymbol{x}) \left[(R(\boldsymbol{x})(\boldsymbol{x} - \boldsymbol{x}_0)) \cdot \boldsymbol{\nabla} \psi^n_{j\boldsymbol{k}}(\boldsymbol{x}) + [\boldsymbol{\nabla} \cdot (R(\boldsymbol{x})(\boldsymbol{x} - \boldsymbol{x}_0))] \overline{\psi^n_{j\boldsymbol{k}}(\boldsymbol{x})} \right] d\boldsymbol{x}$$

which can be split in the sum of two terms r_1 and r_2. The first half of this integral is

$$r_1 = 2^j \iint I(0;\boldsymbol{x}) \, (R(\boldsymbol{x})(\boldsymbol{x} - \boldsymbol{x}_0)) \cdot \frac{\boldsymbol{\nabla} \psi^n_{j\boldsymbol{k}}(\boldsymbol{x})}{2^j} d\boldsymbol{x}$$

Since $2^{-j} \partial \psi^n_{j\boldsymbol{k}}(\boldsymbol{x}) / \partial x_\ell$ are wavelets of a wavelet frame with $\lfloor \alpha + 1 \rfloor$ vanishing moments, and since $\boldsymbol{x} \mapsto I(0;\boldsymbol{x}) R(\boldsymbol{x})(\boldsymbol{x} - \boldsymbol{x}_0)$ is $\alpha + 1$–Lipschitz, Jaffard's theorem [Jaf91, Mal97] implies that

$$\left| \iint I(0;\boldsymbol{x}) \, (R(\boldsymbol{x})(\boldsymbol{x} - \boldsymbol{x}_0)) \cdot \frac{\boldsymbol{\nabla} \psi^n_{j\boldsymbol{k}}(\boldsymbol{x})}{2^j} d\boldsymbol{x} \right| \leq M 2^{-j(\alpha+2)} \left(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1} \right)$$

thus

$$|r_1| \leq M 2^{-j(\alpha+1)} \left(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1} \right) \tag{5.14}$$

The second term r_2 is

$$\iint I(0;\boldsymbol{x}) \, [\boldsymbol{\nabla} \cdot (R(\boldsymbol{x})(\boldsymbol{x} - \boldsymbol{x}_0))] \overline{\psi^n_{j\boldsymbol{k}}(\boldsymbol{x})} d\boldsymbol{x}$$

The left part of the integrand

$$I(0;\boldsymbol{x}) \, [\boldsymbol{\nabla} \cdot (R(\boldsymbol{x})(\boldsymbol{x} - \boldsymbol{x}_0))]$$

being α–Lipschitz and the wavelets $\psi^n_{j\boldsymbol{k}}(\boldsymbol{x})$ having $\lfloor \alpha \rfloor$ vanishing moments, we use again Jaffard's theorem to state that

$$|r_2| \leq M' 2^{-j(\alpha+1)} \left(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha} \right) \tag{5.15}$$

Finally, we combine (5.14) and (5.15) to state that there exists a bound M such that

$$|r(\psi^n, j, \boldsymbol{k})| \leq M 2^{-j(\alpha+1)} \left(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1} \right)$$

To prove the existence of sequences j_p, $\boldsymbol{k}_p$ and n_p that fulfill Eq. (5.13), we use the converse statement of Jaffard's theorem we apply on a sequence of balls around $\boldsymbol{x}_0$ whose radius converges to 0, and notice that $\boldsymbol{x} \mapsto I(0;\boldsymbol{x})$ is not $(\alpha + \alpha')/2$–Lipschitz on $\boldsymbol{x}_0$. □

This lemma shows that around a Lipschitz singularity of order α, the real optic flow fulfills a system of equations of the form:

$$M_{j\boldsymbol{k}}\boldsymbol{v}(\boldsymbol{x}_0) + E_{j\boldsymbol{k}} = Y_{j\boldsymbol{k}} \tag{5.16}$$

where for a chosen sequence j_p, $\boldsymbol{k}_p$ such that j_p tends to $+\infty$ and $2^{-j_p}\boldsymbol{k}_p$ goes to $\boldsymbol{x}_0$, the ratio

$$\begin{aligned}
\frac{\|E_{j_p\boldsymbol{k}_p}\|}{\|M_{j_p\boldsymbol{k}_p}\|} &\leq \frac{M}{N}2^{-j(\alpha+1-\alpha')}\frac{1+|\boldsymbol{k}_p - 2^{j_p}\boldsymbol{x}_0|^{\alpha+1}}{1+|\boldsymbol{k}_p - 2^{j_p}\boldsymbol{x}_0|^{\alpha'}} \\
&\leq \frac{M}{N}2^{-j_p(\alpha+1-\alpha')}\left(1+\frac{|\boldsymbol{k}_p - 2^{j_p}\boldsymbol{x}_0|^{\alpha+1}}{1+|\boldsymbol{k}_p - 2^{j_p}\boldsymbol{x}_0|^{\alpha'}}\right) \\
&\leq \frac{M}{N}2^{-j_p(\alpha+1-\alpha')}\left(1+|\boldsymbol{k}_p - 2^{j_p}\boldsymbol{x}_0|^{\alpha+1-\alpha'}\right) \\
&= \frac{M}{N}\left(2^{-j_p(\alpha+1-\alpha')}+|2^{-j_p}\boldsymbol{k}_p - \boldsymbol{x}_0|^{\alpha+1-\alpha'}\right)
\end{aligned}$$

and thus goes to 0 as p goes to infinity if α' is in the open interval $(\alpha, \alpha+1)$. We have thus proved the first part of Th. 4.1.

We now prove the second part of this theorem:

Theorem 4.1–part 2

If there exists an increasing integer sequence $q \mapsto p(q)$ such that the square matrices $M_{j_{p(q)}\boldsymbol{k}_{p(q)}}$ have a bounded condition number as $q \to +\infty$, then the velocities $\boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}}$ defined as a the solutions of the systems

$$M_{j_{p(q)}\boldsymbol{k}_{p(q)}}\boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}} = Y_{j_{p(q)}\boldsymbol{k}_{p(q)}} \tag{5.17}$$

converge to the true velocity $\boldsymbol{v}(\boldsymbol{x}_0)$.

Proof. We write the difference of equations (5.16) and (5.17), to get

$$M_{j_{p(q)}\boldsymbol{k}_{p(q)}}(\boldsymbol{v}(\boldsymbol{x}_0) - \boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}}) = -E_{j_{p(q)}\boldsymbol{k}_{p(q)}}$$

and denoting "cond(M)" the condition number of a matrix M, we have

$$\|\boldsymbol{v}(\boldsymbol{x}_0) - \boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}}\| \leq \mathrm{cond}(M_{j_{p(q)}\boldsymbol{k}_{p(q)}})\frac{\|E_{j_{p(q)}\boldsymbol{k}_{p(q)}}\|}{\|M_{j_{p(q)}\boldsymbol{k}_{p(q)}}\|}$$

and the estimation error thus goes to zero. □

This result can be extended to the more realistic case when the $M_{j\boldsymbol{k}}$ matrices are not square and have more rows than columns. In this case, the condition number is supposed to be the condition number with respect to left pseudo-inversion, i.e. $\mathrm{cond}(M) = \sqrt{\mathrm{cond}(M^TM)}$, and the solution $\boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}}$ of system (5.17) is the least square solution of the system:

$$\boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}} = \left(M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}}M_{j_{p(q)}\boldsymbol{k}_{p(q)}}\right)M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}}Y_{j_{p(q)}\boldsymbol{k}_{p(q)}}$$

so we state

Theorem 4.1–part 3

If there exists an increasing integer sequence $q \mapsto p(q)$ such that the matrices $M_{j_{p(q)}\boldsymbol{k}_{p(q)}}$ have a bounded condition number with respect to left inversion as $q \to +\infty$, then the velocities $\boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}}$ defined as the least square solutions of the systems

$$M_{j_{p(q)}\boldsymbol{k}_{p(q)}} \boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}} = Y_{j_{p(q)}\boldsymbol{k}_{p(q)}} \tag{5.18}$$

converge to the true velocity $\boldsymbol{v}(\boldsymbol{x}_0)$.

Proof. We write it in a similar way to the proof of the second part of Th. 4.1. By multiplying (5.16) on the left with $M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}}$, we get

$$M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}} M_{j_{p(q)}\boldsymbol{k}_{p(q)}} \boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}} + M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}} E_{j_{p(q)}\boldsymbol{k}_{p(q)}} = M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}} Y_{j_{p(q)}\boldsymbol{k}_{p(q)}}$$

and by difference with (5.18)

$$M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}} M_{j_{p(q)}\boldsymbol{k}_{p(q)}} \left(\boldsymbol{v}(\boldsymbol{x}_0) - \boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}}\right) = -M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}} E_{j_{p(q)}\boldsymbol{k}_{p(q)}}$$

This gives the bound

$$\left\|\boldsymbol{v}(\boldsymbol{x}_0) - \boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}}\right\| \leq \mathrm{cond}(M_{j_{p(q)}\boldsymbol{k}_{p(q)}})^2 \frac{\|M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}} E_{j_{p(q)}\boldsymbol{k}_{p(q)}}\|}{\|M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}} M_{j_{p(q)}\boldsymbol{k}_{p(q)}}\|}$$

and since

$$\left\{\begin{array}{rcl} \|M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}} E_{j_{p(q)}\boldsymbol{k}_{p(q)}}\| & \leq & \|M_{j_{p(q)}\boldsymbol{k}_{p(q)}}\| \times \|E_{j_{p(q)}\boldsymbol{k}_{p(q)}}\| \\ \|M^T_{j_{p(q)}\boldsymbol{k}_{p(q)}} M_{j_{p(q)}\boldsymbol{k}_{p(q)}}\| & = & \|M_{j_{p(q)}\boldsymbol{k}_{p(q)}}\|^2 \end{array}\right.$$

we get

$$\left\|\boldsymbol{v}(\boldsymbol{x}_0) - \boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}}\right\| \leq \mathrm{cond}(M_{j_{p(q)}\boldsymbol{k}_{p(q)}})^2 \frac{\|E_{j_{p(q)}\boldsymbol{k}_{p(q)}}\|}{\|M_{j_{p(q)}\boldsymbol{k}_{p(q)}}\|}$$

Thus

$$\boldsymbol{v}_{j_{p(q)}\boldsymbol{k}_{p(q)}} \to \boldsymbol{v}(\boldsymbol{x}_0) \qquad \text{as } q \to +\infty$$

□

The results stated in Th. 4.1 are weak because they only state that a subsequence of the linear systems we solve is able to give us a proper estimate of the flow. It is however hardly possible to give a much stronger result, as the next example (or counter example) demonstrates it in 1D.

Let us a consider a pattern $I(x)$ that is 1.5–Lipschitz on 0: $I(x) = x\sqrt{|x|}$, and a velocity flow $v(x) = v_0 + ax$ for some constants a and v_0. Let us suppose we use a real odd wavelet

ψ (i.e. $\psi(-x) = -\psi(x)$). The 1×1 "matrix" M_{jk} for any j and $k = 0$ is $M_{j0} = 0$ because it is the integral of an odd function. The estimation error is however

$$\begin{aligned} E_{j0} &= \int x\sqrt{|x|}\psi_{j0}dx \\ &= a2^{j/2}\int x\sqrt{|x|}\psi(2^j x)dx \\ &= a2^{-j/2}\int 2^{-j}x\sqrt{|2^{-j}x|}\psi(x)dx \\ &= a2^{-2j}\int x|x|^{1/2}\psi(x)dx \\ &= a2^{-2j}E_{00} \end{aligned}$$

There exists some odd wavelet ψ such that E_{00} is nonzero, which implies that for all j, the ratio $\|E_{j0}\|/\|M_{j0}\|$ is infinite. So we cannot ever dream of using one of these measures to estimate the flow. The above theorem is valid, however, because the measurement is possible thanks to lateral measures with small nonzero indices k.

Up to now, we have studied the estimation error coming from the approximation that the optic flow is locally constant. As a second step, we now focus on time aliasing.

5.2.2 Error caused by time aliasing

Time aliasing error is caused by the approximation of the picture time derivative with a finite time difference. To limit the number of equations and to avoid lengthy proofs, we limit ourselves to the simpler study of the case of the first order approximation of the time derivative

$$\frac{\partial I}{\partial t} \simeq \frac{I(t+\delta t) - I(t)}{\delta t}$$

and not of our real implementation (see Eqs. (3.5) (3.6)), but the proofs are the same in spirit.

We begin by stating a couple of lemmata. Lemma 5.3 focuses on the impact on the wavelet coefficients of a given function of a smooth deformation of this function and concentrates the most technical part of this section. Lemma 5.4 states the existence of an approximate linear system and has the same purpose as lemma 5.2.

Lemma 5.3

Let $(t, \boldsymbol{x}) \mapsto X_t(\boldsymbol{x})$ be a $\lfloor\alpha\rfloor + 3$ times continuously differentiable mapping of $\mathbb{R} \times \mathbb{R}^2 \to \mathbb{R}^2$ such that $\boldsymbol{x} \mapsto X_t(\boldsymbol{x})$ is invertible for any t, and such that $X_0(\boldsymbol{x}) = \boldsymbol{x}$ for any $\boldsymbol{x} \in \mathbb{R}^2$.

Let f be some function $\mathbb{R}^2 \to \mathbb{R}$ that is α–Lipschitz at $\boldsymbol{x}_0$. Let $(\Psi^n_{j\boldsymbol{k}})$ be a wavelet frame with $\lfloor\alpha\rfloor + 1$ vanishing moments. There exist $M > 0$, $j_0 \in \mathbb{Z}$, $\eta > 0$ and $t_0 > 0$ such that

$$\iint \left(f(X_t(\boldsymbol{x})) - f(\boldsymbol{x})\right)\Psi^n_{j\boldsymbol{k}}d\boldsymbol{x} \leq M\left(2^{-j(\alpha+2)} + |t|2^{-j\alpha}\left(1 + |2^j\boldsymbol{x}_0 - \boldsymbol{k}|^\alpha\right)\right) \tag{5.19}$$

for $j > j_0$, $|2^{-j}\boldsymbol{k} - \boldsymbol{x}_0| < \eta$ and $|t| \leq t_0 2^{-j}$.

Proof. Since f is α–Lipschitz at $\boldsymbol{x}_0$, we can write

$$f(\boldsymbol{x}) = P(\boldsymbol{x}) + r(\boldsymbol{x})$$

where P is a polynomial of degree less than α and $|r(\boldsymbol{x})| \leq M|\boldsymbol{x} - \boldsymbol{x}_0|^\alpha$ for some $M \in [0, +\infty)$.

The left hand side of (5.19) can then be written

$$\iint \left(f(X_t(\boldsymbol{x})) - f(\boldsymbol{x})\right) \Psi^n_{j\boldsymbol{k}}(\boldsymbol{x}) d\boldsymbol{x} = \mathcal{A} + \mathcal{B}$$

where

$$\mathcal{A} = \iint \left(P(X_t(\boldsymbol{x})) - P(\boldsymbol{x})\right) \Psi^n_{j\boldsymbol{k}}(x) d\boldsymbol{x}$$

and

$$\mathcal{B} = \iint \left(r(X_t(\boldsymbol{x})) - r(\boldsymbol{x})\right) \Psi^n_{j\boldsymbol{k}}(\boldsymbol{x}) d\boldsymbol{x}$$

Since Ψ^n has $\lfloor \alpha \rfloor + 1$ vanishing moments,

$$\mathcal{A} = \iint P(X_t(\boldsymbol{x})) \Psi^n_{j\boldsymbol{k}}(x) d\boldsymbol{x}$$

Since the mapping $\boldsymbol{x} \mapsto P(X_t(\boldsymbol{x}))$ is uniformly $\alpha + 3$–Lipschitz, it is also uniformly $\alpha + 1$–Lipschitz, which implies

$$\mathcal{A} \leq M' 2^{-j(\alpha+2)} \tag{5.20}$$

If we now focus on bounding $\mathcal{B}$, we write

$$\mathcal{B} = \iint r(X_t(\boldsymbol{x})) \Psi^n_{j\boldsymbol{k}}(\boldsymbol{x}) d\boldsymbol{x} - \iint r(\boldsymbol{x}) \Psi^n_{j\boldsymbol{k}}(\boldsymbol{x}) d\boldsymbol{x}$$

With a variable change $\boldsymbol{x} \mapsto X_t^{-1}(\boldsymbol{x})$, and defining

$$K(\boldsymbol{x}, t) = \det\left(\frac{\partial X_t^{-1}}{\partial \boldsymbol{x}}(\boldsymbol{x}, t)\right)$$

we get

$$\mathcal{B} = \iint \left(K(\boldsymbol{x}, t) \Psi^n_{j\boldsymbol{k}}(X_t^{-1}(\boldsymbol{x})) - I_2 \Psi^n_{j\boldsymbol{k}}(\boldsymbol{x})\right) r(\boldsymbol{x}) d\boldsymbol{x}$$

Note that K is $\lfloor \alpha \rfloor + 2$ times continuously differentiable, and that we also have $K(\boldsymbol{x}, 0) = 1$ for all $\boldsymbol{x}$. $\mathcal{B}$ can again be split into a sum of two terms:

$$\begin{array}{rcccc} \mathcal{B} & = & \iint K(\boldsymbol{x}, t) \left(\Psi^n_{j\boldsymbol{k}}(X_t^{-1}(\boldsymbol{x})) - \Psi^n_{j\boldsymbol{k}}(\boldsymbol{x})\right) r(\boldsymbol{x}) d\boldsymbol{x} & + & \iint (K(\boldsymbol{x}, t) - 1) \Psi^n_{j\boldsymbol{k}}(\boldsymbol{x}) r(\boldsymbol{x}) d\boldsymbol{x} \\ & = & \mathcal{B}_1 & + & \mathcal{B}_2 \end{array}$$

$\mathcal{B}_2$ is bounded by

$$\begin{aligned}
\mathcal{B}_2 &\leq |t|M'' \iint |\Psi^n_{j\boldsymbol{k}}(\boldsymbol{x})| \times |\boldsymbol{x}-\boldsymbol{x}_0|^\alpha d\boldsymbol{x} && (5.21)\\
&= |t|M''2^{-j} \iint |2^{-j}\boldsymbol{y}+2^{-j}\boldsymbol{k}-\boldsymbol{x}_0|^\alpha \Psi^n(\boldsymbol{y})d\boldsymbol{y} && (5.22)\\
&\leq 2^\alpha |t|M''2^{-j(\alpha+1)} \left(\iint |\boldsymbol{y}|^\alpha |\Psi^n(\boldsymbol{y})|d\boldsymbol{y} + |2^j\boldsymbol{x}_0-\boldsymbol{k}|^\alpha \iint |\Psi^n(\boldsymbol{y})|d\boldsymbol{y} \right) && (5.23)\\
&\leq |t|M'''2^{-j(\alpha+1)} \left(1+|2^j\boldsymbol{x}_0-\boldsymbol{k}|^\alpha\right)
\end{aligned}$$

where (5.21) leads to (5.22) through a variable change $\boldsymbol{y}=2^j\boldsymbol{x}-\boldsymbol{k}$ and to (5.23) by noticing that $(a+b)^\alpha \leq 2^\alpha(a^\alpha+b^\alpha)$ for $a>0$, $b>0$.

A similar computation gives a bound on $\mathcal{B}_1$:

$$\begin{aligned}
\mathcal{B}_1 &\leq M \iint |\Psi^n_{j\boldsymbol{k}}(X_t^{-1}(\boldsymbol{x})) - \Psi^n_{j\boldsymbol{k}}(\boldsymbol{x})| \times |\boldsymbol{x}-\boldsymbol{x}_0|^\alpha d\boldsymbol{x}\\
&= M2^j \iint \left|\Psi^n(2^jX_t^{-1}(\boldsymbol{x})-\boldsymbol{k}) - \Psi^n(2^j\boldsymbol{x}-\boldsymbol{k})\right| \times |\boldsymbol{x}-\boldsymbol{x}_0|^\alpha d\boldsymbol{x}\\
&= M2^{-j(\alpha+1)} \iint \left|\Psi^n\left(2^jX_t^{-1}(2^{-j}(\boldsymbol{y}+\boldsymbol{k}))\right) - \Psi^n(\boldsymbol{y})\right| \times |\boldsymbol{y}+\boldsymbol{k}-2^j\boldsymbol{x}_0|^\alpha d\boldsymbol{y}
\end{aligned}$$

We first notice that if $t \leq t_0 2^{-j}$,

$$\boldsymbol{y} \mapsto |\Psi^n\left(2^jX_t^{-1}(2^{-j}(\boldsymbol{y}+\boldsymbol{k}))-\boldsymbol{k}\right) - \Psi^n(\boldsymbol{y})|$$

is of bounded support. Second, if the support of the wavelet $\Psi^n_{j\boldsymbol{k}}$ is in some vicinity of $\boldsymbol{x}_0$ (which is the same as requiring that j be large enough and $|2^{-j}\boldsymbol{k}-\boldsymbol{x}_0|$ be small enough), we can write

$$|\Psi^n\left(2^jX_t^{-1}(2^{-j}(\boldsymbol{y}+\boldsymbol{k}))-\boldsymbol{k}\right) - \Psi^n(\boldsymbol{y})| \leq M2^j|X_t^{-1}(2^{-j}(\boldsymbol{y}+\boldsymbol{k})) - 2^{-j}(\boldsymbol{y}+\boldsymbol{k})| \quad \leq M'2^j|t|$$

for some M' so that we finally get by integration

$$\mathcal{B}_1 \leq M''2^{-j\alpha}|t|\left(1+|2^j\boldsymbol{x}_0-\boldsymbol{k}|^\alpha\right)$$

and for $\mathcal{B}$ the upper bound

$$\mathcal{B} \leq M2^{-j\alpha}|t|\left(1+|2^j\boldsymbol{x}_0-\boldsymbol{k}|^\alpha\right) \tag{5.24}$$

for some other M.

To summarize, we combine the bounds on $\mathcal{A}$ and $\mathcal{B}$ which implies (5.19):

$$\iint \left(f(X_t(\boldsymbol{x}))-f(\boldsymbol{x})\right)\Psi^n_{j\boldsymbol{k}}d\boldsymbol{x} \leq M\left(2^{-j(\alpha+2)} + |t|2^{-j\alpha}\left(1+|2^j\boldsymbol{x}_0-\boldsymbol{k}|^\alpha\right)\right) \tag{5.19}$$

□

Lemma 5.4

Let α be a positive real. Assume that $\boldsymbol{x} \mapsto I(0;\boldsymbol{x})$ is exactly α–Lipschitz at $\boldsymbol{x}_0$. There exist $j_0 \in \mathbb{Z}$, $t_0 > 0$, $C > 0$ and $M \geq 0$ such that if $j > j_0$, $\delta t \in [0, 2^{-j}t_0]$ and $|\boldsymbol{x}_0 - 2^{-j}\boldsymbol{k}| \leq C$ then

$$v_1(\boldsymbol{x}_0,0)\left\langle I(0), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_1}\right\rangle + v_2(\boldsymbol{x}_0,0)\left\langle I(0), \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_2}\right\rangle + r'(\psi^n, j, \boldsymbol{k}) = \left\langle \frac{I(\delta t) - I(0)}{\delta t}, \psi^n_{j\boldsymbol{k}}\right\rangle \tag{5.25}$$

where

$$|r'(\psi^n, j, \boldsymbol{k})| \leq M\left(2^{-j(\alpha+1)}(1 + |2^j\boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1}) + |\delta t|2^{-j(\alpha-1)}(1 + |2^j\boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha})\right) \tag{5.26}$$

Proof. Lemma 5.2 can be applied to the frame $I(t)$ at any time t, where the position $\boldsymbol{x}_0$ has to be replaced with the new location where I is α–Lipschitz, i.e. $\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0, t)$. We can thus write Eq. (5.11) for any time t as:

$$\boldsymbol{v}(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0,t),t) \cdot \iint \boldsymbol{\nabla} I(t;\boldsymbol{x})\psi^n_{j\boldsymbol{k}} d\boldsymbol{x} + r(\psi^n, j, \boldsymbol{k}, t) = \left\langle \frac{\partial I}{\partial t}(t), \psi^n_{j\boldsymbol{k}}\right\rangle$$

where

$$|r(\psi^n, j, \boldsymbol{k}, t)| \leq M2^{-j(\alpha+2)}(1 + |2^j(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0,t)) - \boldsymbol{k}|^{\alpha})$$

In this inequality, one can suppose that M can be taken independent of t over some bounded time and space domain around $\boldsymbol{x}_0$.

This relationship can be integrated over time to get

$$\int_0^{\delta t} \boldsymbol{v}(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0,t),t) \cdot \iint \boldsymbol{\nabla} I(t;\boldsymbol{x})\psi^n_{j\boldsymbol{k}} d\boldsymbol{x}dt + \int_0^{\delta t} r(\psi^n, j, \boldsymbol{k}, t) = \langle I(\delta t) - I(0), \psi^n_{j\boldsymbol{k}}\rangle$$

We can now write Eq. (5.25)

$$\boldsymbol{v}(\boldsymbol{x}_0) \iint \boldsymbol{\nabla} I(0)\psi^n_{j\boldsymbol{k}} d\boldsymbol{x} + r'(\psi^n, j, \boldsymbol{k}, \delta t) = \left\langle \frac{I(\delta t) - I(0)}{\delta t}, \psi^n_{j\boldsymbol{k}}\right\rangle$$

where the residual r' can be expressed as

$$r' = \frac{1}{\delta t}\int_0^{\delta t} r(\psi^n, j, \boldsymbol{k}, t)dt + \frac{1}{\delta t}\int_0^{\delta t}\left(\boldsymbol{v}(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0,t),t)\iint \boldsymbol{\nabla} I(t;\boldsymbol{x})\psi^n_{j\boldsymbol{k}}(\boldsymbol{x})d\boldsymbol{x}\right.$$
$$\left. - \boldsymbol{v}(\boldsymbol{x}_0,0)\iint \boldsymbol{\nabla} I(0;\boldsymbol{x})\psi^n_{j\boldsymbol{k}}(\boldsymbol{x})d\boldsymbol{x}\right)dt$$

Our purpose is now to bound properly this residual. We write it $r' = r'_1 + r'_2$. The first part r'_1 is easily settled:

$$|r(\psi^n, j, \boldsymbol{k}, t)| \leq M2^{-j(\alpha+1)}(1 + |2^j(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0,t)) - \boldsymbol{k}|^{\alpha+1}) \qquad \text{by Lemma 5.2}$$

Noticing that $|\boldsymbol{\delta}(\boldsymbol{x},t)| \leq M'|t|$ and that $(a+b)^\alpha \leq 2^\alpha(a^\alpha + b^\alpha)$ for any $a, b > 0$ and $\alpha > 0$, we obtain

$$\begin{aligned}|r(\psi^n, j, \boldsymbol{k}, t)| &\leq M' 2^{-j(\alpha+1)} \left(1 + |2^j t|^{\alpha+1} + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1}\right) \\ &\leq M'' \left(|t|^{\alpha+1} + 2^{-j(\alpha+1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1})\right)\end{aligned}$$

by integration, we get

$$\begin{aligned}|r_1'| &\leq \frac{1}{\delta t} \int_0^{\delta t} |r(\psi^n, j, \boldsymbol{k}, t)| dt \\ &\leq M''' \left(|t|^{\alpha+1} + 2^{-j(\alpha+1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1})\right)\end{aligned}$$

Since we assume that $|t| \leq t_0 2^{-j}$, we can simplify this bound as

$$|r_1'| \leq M 2^{-j(\alpha+1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1}) \tag{5.27}$$

The second part r_2' is

$$r_2' = \frac{1}{\delta t} \int_0^{\delta t} \left(\boldsymbol{v}(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0, t), t) \iint \boldsymbol{\nabla} I(t; \boldsymbol{x}) \psi_{j\boldsymbol{k}}^n(\boldsymbol{x}) d\boldsymbol{x} - \boldsymbol{v}(\boldsymbol{x}_0, 0) \iint \boldsymbol{\nabla} I(0; \boldsymbol{x}) \psi_{j\boldsymbol{k}}^n(\boldsymbol{x}) d\boldsymbol{x} \right) dt$$

The integrand can be written

$$\boldsymbol{v}(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0, t), t) \iint (\boldsymbol{\nabla} I(t; \boldsymbol{x}) - \boldsymbol{\nabla} I(0; \boldsymbol{x})) \, \psi_{j\boldsymbol{k}}^n d\boldsymbol{x} \tag{$\mathcal{C}$}$$

$$+ (\boldsymbol{v}(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0, t), t) - \boldsymbol{v}(\boldsymbol{x}_0, 0)) \iint \boldsymbol{\nabla} I(0; \boldsymbol{x}) \psi_{j\boldsymbol{k}}^n d\boldsymbol{x} \tag{$\mathcal{D}$}$$

With an integration by parts

$$\mathcal{D} = -(\boldsymbol{v}(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0, t), t) - \boldsymbol{v}(\boldsymbol{x}_0, 0)) \iint I(0; \boldsymbol{x}) \boldsymbol{\nabla} \psi_{j\boldsymbol{k}}^n d\boldsymbol{x}$$

We notice that $(2^{-j} \partial \psi_{j\boldsymbol{k}}^n / \partial x_\ell)_{j\boldsymbol{k}\ell n}$ make up a frame of $L_2(\mathbb{R}^2)$, and that $I(0)$ is α–Lipschitz at $\boldsymbol{x}_0$, so that thanks to Jaffard's theorem,

$$|\mathcal{D}| \leq M|t| 2^{-j\alpha}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^\alpha)$$

To bound the term $\mathcal{C}$, we write with an integration by parts

$$\begin{aligned}\mathcal{C} &= -\boldsymbol{v}(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0, t), t) \iint (I(t; \boldsymbol{x}) - I(0; \boldsymbol{x})) \, \boldsymbol{\nabla} \psi_{j\boldsymbol{k}}^n d\boldsymbol{x} \\ &= -2^j \boldsymbol{v}(\boldsymbol{x}_0 + \boldsymbol{\delta}(\boldsymbol{x}_0, t), t) \iint (I(0; \boldsymbol{x} - \boldsymbol{\epsilon}(\boldsymbol{x}, t)) - I(0; \boldsymbol{x})) \, \frac{\boldsymbol{\nabla} \psi_{j\boldsymbol{k}}^n}{2^j} d\boldsymbol{x}\end{aligned}$$

so that we can apply Lemma 5.3 with $f = I$, and with

$$\Psi_{j\boldsymbol{k}}^{(n,\ell)} = 2^{-j} \frac{\partial \psi_{j\boldsymbol{k}}^n}{\partial x_\ell}$$

to finally obtain

$$|\mathcal{C}| \leq M' \left(2^{-j(\alpha+1)} + |t|2^{-j(\alpha-1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^\alpha)\right)$$

By integration,

$$\begin{aligned} r'_2 &= \frac{1}{\delta t} \int_0^{\delta t} \mathcal{C} + \mathcal{D} dt \\ r'_2 &\leq M \left(2^{-j(\alpha+1)} + |\delta t|2^{-j(\alpha-1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^\alpha)\right) \end{aligned} \tag{5.28}$$

so that combining (5.27) and (5.28), we can bound r' completely:

$$\begin{aligned} |r'| &\leq |r'_1| + |r'_2| \\ &\leq M \left(2^{-j(\alpha+1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1}) + |\delta t|2^{-j(\alpha-1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^\alpha)\right) \end{aligned}$$

□

We can write Eq. (5.25) in short:

$$M_{j\boldsymbol{k}} \boldsymbol{v}(\boldsymbol{x}_0, 0) + E'_{j\boldsymbol{k}} = Y'_{j\boldsymbol{k}}$$

where the matrix $M_{j\boldsymbol{k}}$ is the same as in (5.16). We just showed that

$$\|E'_{j\boldsymbol{k}}\| \leq M \left(2^{-j(\alpha+1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1}) + |\delta t|2^{-j(\alpha-1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^\alpha)\right)$$

We can now proceed with the proof of the first part of Th. 4.2. With the same argument as in Th. 4.1, for $\alpha' > \alpha$, there exists a sequence j_p, $\boldsymbol{k}_p$ such that

$$\|M_{j_p \boldsymbol{k}_p}\| \geq N 2^{-j_p \alpha'} \left(1 + |2^{j_p} \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha'}\right)$$

The error ratio can be bounded with

$$\frac{\|E_{j_p \boldsymbol{k}_p}\|}{\|M_{j_p \boldsymbol{k}_p}\|} \leq \frac{M}{N} \left(\frac{2^{-j(\alpha+1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha+1})}{2^{-j\alpha'}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^\alpha)} + |\delta t| \frac{2^{-j(\alpha-1)}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^\alpha)}{2^{-j\alpha'}(1 + |2^j \boldsymbol{x}_0 - \boldsymbol{k}|^{\alpha'})}\right)$$

Let θ be any strictly positive real. We choose a sequence δt_p such that

$$\delta t_p \leq 2^{-j(1+\theta)}. \tag{5.29}$$

We take $\alpha' \in (\alpha, \alpha + \theta/2) \cap (\alpha, \alpha + 1)$, and there exists some p_0 and M' such that $p \geq p_0$ implies:

$$\frac{\|E_{j_p \boldsymbol{k}_p}\|}{\|M_{j_p \boldsymbol{k}_p}\|} \leq M' |\delta t_p| 2^{-j(\alpha-1-\alpha')}$$

and vanishes as $p \to +\infty$. The first part of Th. 4.2 is thus proved. The last part is proved in exactly the same way as Th. 4.1–parts 2 and 3, so it is not repeated here.

Bibliography

[Coh92] A. Cohen. *Ondelettes et traitement numérique du signal*. Masson, 1992.

[Jaf91] S. Jaffard. Pointwise smoothness, two-microlocalization and wavelet coefficients. *Publicaciones Matemáticas*, 1:316–328, 1991.

[Mal97] S. G. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1997.

Chapter 6

Numerical experimentation and prospects

6.1 Numerical experimentation

The algorithm was implemented with a dedicated set of *analytic* mother wavelets. The motivation of their use as well as their construction are described in section 4.2. We shortly describe some implementation issues and detail the numerical validation of our algorithm.

Filters The filters used throughout all experimentations are usually the same: we use Deslauriers–Dubuc filters of varying orders. The low–pass Deslauriers–Dubuc filters are

$$\begin{bmatrix} -1 & 0 & 9 & 16 & 9 & 0 & -1 \end{bmatrix}/16 \tag{6.1a}$$

or

$$\begin{bmatrix} 3 & 0 & -25 & 0 & 150 & 256 & 150 & 0 & -25 & 0 & 3 \end{bmatrix}/256 \tag{6.1b}$$

These filters have several advantages. They have a good frequency resolution (and number of vanishing moments) with respect to the number of operations for a given convolution: they bypass the classical limitation according to which the number of vanishing moments is bounded for a given filter *support size* and a given computational complexity because they have a number of *zero places* that are inside the support without costing additional flops. They have a second advantage we did not benefit from in our implementation: all their coefficients are dyadic rationals, and the related computations can be limited to integer computations. The wavelet transforms were always oversampled with a factor of 2, so that the alias free measurement range in (3.9) was multiplied with a factor of 2.

The corresponding high–pass filters m_1 are made from these low pass–filters with a π frequency shift. The analytic filter is obtained from the low–pass filter m_0 with a $\pi/2$ frequency shift. Also note that a convolution with this complex filter is of the same complexity as with a real filter, because its coefficients are either real or of vanishing real part.

Dealing with boundary effects Several methods were tested to reduce the loss of accuracy caused by the boundary occlusions: zero, symmetric or constant padding. The best error reduction (based on experimental comparisons) are achieved by continuous extensions of the picture after the boundaries: either symmetric or constant padding, as already noticed by Simoncelli [Sim98]. To limit the increase of computations and storage requirements, these constant paddings are done at the level of the convolution function at any resolution. A signal is automatically padded with constant extensions as required by the support of the filter.

6.1.1 Computational cost

We claimed that our algorithm computes an optic flow map in $\mathcal{O}(N)$ flops where N is the number of pixels in the picture. In this section we estimate the constant A such that the cost is not more than $A \times N$. The computational cost has been estimated for two situations: with or without estimation of illumination changes. The cost in flops per pixel includes prefiltering, pyramid filtering and subsampling and system solving, frequency shift correction and interpolation from the resolution of 2×2 to that of 1×1 pixel. Similar complexity estimates from Magarey and Kingsbury [MK98] are given for comparison in table 6.1. A small difference is that Magarey and Kingsbury estimated the complexity of their algorithm on 5 pyramid levels, whereas our estimate is a bound valid for any number of levels.

Method	complexity
Magarey & Kingsbury	1618
This (with illumination)	863
This (without illumination)	780

Table 6.1: Complexity in FLOPS per pixel

6.1.2 True sequences

Image sequences were downloaded from Barron *& al.*'s FTP site at *csd.uwo.ca*. The algorithm was tested on the rubic sequence (a rubic's cube on a rotating plate), the taxi sequence (where three vehicles are moving respectively towards East, West and Northwest). One frame from the sequence as well as a map of the measured optic flow are displayed in Fig. 6.1.2 and 6.1.2.

6.1.3 Synthetic sequences

The described algorithm was also run on classical synthetic sequences (including Yosemite), and the result was compared to other methods. The error measurement was done in terms of angular error, as introduced by Fleet & Jepson in [FJ90]. The "distance" between a measured velocity $\tilde{\boldsymbol{v}} = (\tilde{v}_1, \tilde{v}_2)$ and a real velocity $\boldsymbol{v} = (v_1, v_2)$ is the angle in degrees

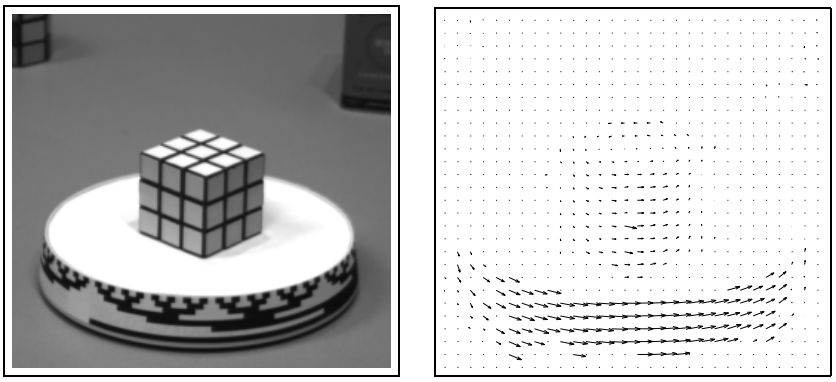


Figure 6.1: Rubic sequence and flow

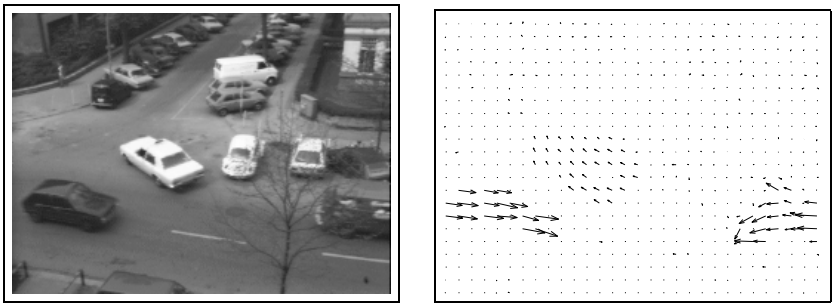


Figure 6.2: Taxi sequence

between the vectors

$$\begin{bmatrix} \tilde{v}_1 \\ \tilde{v}_2 \\ 1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} v_1 \\ v_2 \\ 1 \end{bmatrix}$$

which is $\arccos(1 + \tilde{\boldsymbol{v}} \cdot \boldsymbol{v})$.

For the translating tree sequence, the average angular error is 0.78^o with a density of 99.3%, which can be compared to Magarey & Kingsbury's 1.32^o. Table 6.2 recalls similar measurements done by other authors.

Frames	Method	Avg. Error	Density
21	Fleet & Jepson	0.32^o	74.5%
10	Weber & Malik	0.49^o	96.8%
2	Magarey & Kingsbury	1.32^o	100.0%
2	This method (with illumination)	0.78^o	99.3%

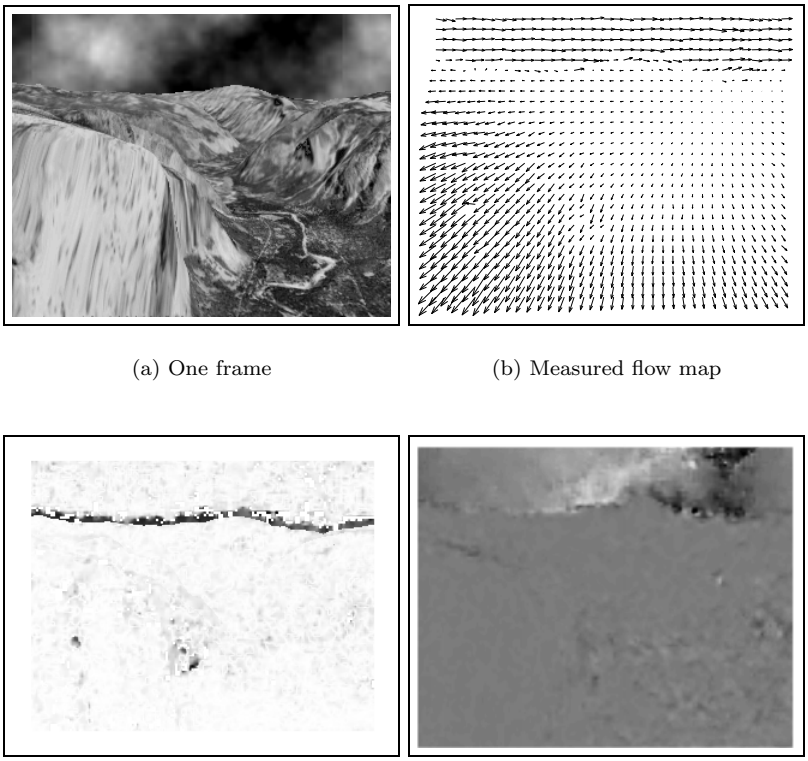
Table 6.2: Compared angular errors for the translating tree sequence. In this table, the first column indicates the number of frames required to perform the measurement. Note that with that respect, the least expensive methods are Magarey and Kingsbury's method, as well as ours.

For the Yosemite sequence, the average angular error over the whole picture (including ground and sky, less 16 pixels on each boundary) is 6.52^o. The measurement density is 96.5%, because some measures were rejected because of aliasing. This result can be compared to Magarey & Kingsbury's 6.20^o. For this sequence, the flow map and angular error map are displayed as well as the map of measured illumination changes are displayed. The average angular errors are displayed in Tab. 6.3.

Other methods [WKCL98, MP98] can provide better accuracy on the optic flow estimation, at the cost of a much higher computation time. Mémin and Pérez [MP98] have an average error of 5.38^o for the Yosemite sequence, and Wu *et al.* [WKCL98] also claim for an a smaller estimation error, but both methods minimize nonconvex functionals and rely on iterative minimization processes.

Frames	Method	Avg. error	Density
21	Fleet & Jepson	4.63^o	34.1%
10	Weber & Malik	4.31^o	64.2%
2	Magarey & Kingsbury	6.20^o	100.0%
2	This method (with illumination)	6.50^o	96.5%

Table 6.3: Compared angular error measures for the Yosemite sequence. The same comments as for Tab. 6.2 hold for these results.



(a) One frame (b) Measured flow map

(c) Angular error map (d) Measured illumination changes

Figure 6.3: Yosemite sequence.We see that the angular error is high on the horizon, which is an occlusion. The illumination change map indicates whether the illumination does not change (in gray), is increased (white shades) or decreases (darker shades). The algorithm detects that the right part of the left cloud is getting lighter, while the left part of the right cloud is getting darker, which corresponds at least to the impression the sequence leaves when it is viewed. We estimate the accuracy of the illumination change measurement in the following section.

6.1.4 Illumination changes

We use the optic flow equation defined in 4. The flow and illumination change estimation is done with an additional measure wavelet on nonzero integral:

$$\Psi^0(x_1, x_2) = \phi(x_1)\phi(x_2) \tag{6.2}$$

All wavelets $\Psi^1, \ldots, \Psi^4$ of eqs.(4.25a–d) used until now to measure the velocity are of zero integral. The purpose of such a setting is the ensure the robustness of the measurement against illumination changes. Now our purpose is to measure these illumination changes, and for this reason we add this new wavelet of nonzero DC response that "see" the overall illumination changes. As a consequence, any local linear system has now an additional unknown L'/L and an additional real linear constraint obtained by inner product with Ψ^0.

Adding a new parameter to explain changes in illumination must first ensure that the flow estimation is robust with respect to global illumination changes. Our algorithm is insensitive to additive pixel value offsets by construction: the wavelets used for OF measurement are of zero DC response. To check to sensitivity of the algorithm to multiplicative pixel value changes, we measured the OF between picture number 20 of the translating tree sequence and picture number 21 multiplied by a varying factor in a very wide range $[0.5, 1.5]$. The results are shown in table 6.4. They are very good compared to classical methods (fractional block matching, gradient pixel recursive) as reported by Magarey & Kingsbury, where a scaling of 0.95 or 1.05 already multiplies the angular error by a factor of 10.

We also compared the average measured illumination change with the actual value: $\Delta L/L$. For wide changes, the actual value is

$$2\frac{L(t+1) - L(t)}{L(t) + L(t+1)}$$

where t and $t+1$ are the times of the two successive frames used for our calculation. This comes from the fact that our algorithm takes as a reference frame to measure the flow and the illumination changes an estimate of the half time frame $I(t+1/2)$ which is estimated with

$$I(t+1/2) \simeq \frac{I(t) + I(t+1)}{2}$$

as detailed in section 3.2 on time aliasing.

As an example, for a multiplication of $I(t+1)$ with a factor of 0.5, we expect the measured illumination change factor to be $2 \times (0.5 - 1)/(1 + 0.5) \simeq -0.67$. Looking at table 6.4, we see that the illumination changes are estimated without bias.

A second test was done on the same sequence (translating tree). Frame 20 was left unchanged, and frame 21 was multiplied with a Gaussian profile

$$g(x, y) = 1 + e^{-\frac{(x-x_0)^2 + (y-y_0)^2}{2\sigma^2}} \tag{6.3}$$

where the center (x_0, y_0) is $(75, 75)$ and $\sigma = 75/2$. We remind that the frame size of the pictures in the translating tree sequence is 150×150 pixels. The relative illumination change

Scaling factor s	0.5	0.7	0.9	1.0	1.1	1.3	1.5
Angular error	1.33^o	0.84^o	0.76^o	0.78^o	0.79^o	0.84^o	0.93^o
Real ill. change $2(s-1)/(1+s)$	-0.67	-0.35	-0.105	0	0.095	0.26	0.4
Measured ill. change	-0.67	-0.35	-0.105	0.00	0.095	0.26	0.40

Table 6.4: Angular error measure, real illumination change and average of measured illumination change for the translating tree sequence. Note that the range of illumination changes that our algorithm can deal with is very wide.

profile is thus expected to be:

$$\frac{L'}{L} \simeq 2\frac{g(x,y)-1}{1+g(x,y)}$$

which has a maximum of $2/3$. The two successive frames of the translating tree sequence are displayed in Fig. 6.4.

Figure 6.4: Two successive frames of the translating tree sequence, where the second frame is multiplied by a Gaussian profile of Eq. (6.3).

The optic flow and an illumination change map were estimated with our algorithm. The average angular error for flow estimation is 2.45^o. The average error in illumination change estimation was estimated in the following way: denoting λ the measured illumination change map that was supposed to estimate L'/L, we computed the L_1 average error

$$E = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}\left|\lambda(i,j) - \frac{L'(i,j)}{L(i,j)}\right|$$

This measured error was 0.0255. Compared to a maximum of 0.67, this is a relative error of 3.8%. Both the true illumination change map and the measured map are displayed in Fig. 6.5.

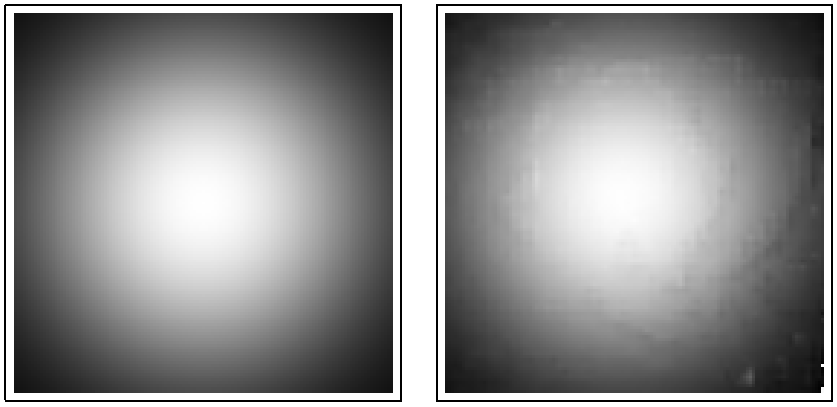


Figure 6.5: Illumination change maps for the modified translating tree sequence. Left: true map and right: measured map.

We think that besides enhancing the stability of the flow measurement with respect to strong illumination changes, the measurement of this additional parameter can be useful in coding video sequences. For a number of reasons, the illumination of a given image sequence can change with time. This can be caused by moving light sources, as well as a single camera motion. If the camera is travelling from a dark area to a lighter one, the illumination or color balancing system changes the local color and illumination of a moving feature in the sequence.

6.2 Video compression

Estimating the optic flow is a way to capture the temporal redundancy between successive frames of a given video sequence, and to reduce the amount of bits to code this sequence. In this spirit, we describe a first simple way to do video compression with motion compensation.

We denote the video sequence by $I(t; \boldsymbol{x})$ with $t \in \mathbb{N}$. A simple way to code such a sequence is to code each of the frames $I(t)$ with some static picture compression algorithm (JPEG, SPIHT, etc.). An elementary way to take advantage of the temporal redundancy of the picture sequence is to code the first frame $I(0)$, and to code next only the differences between successive frames $I(t+1) - I(t)$. This strategy is interesting if a large part of the picture is not moving (like characters moving on a immobile background). It fails however as soon as too large a part of the picture is mobile (like because of the camera motion, with a travelling or a zoom effect).

A second approach consists in doing motion compensation. We measure the displacement $\boldsymbol{v}(t+1/2)$ between the pictures $I(t)$ and $I(t+1)$. If the measure is accurate and if the optic flow can explain the image modifications in time, we can reasonably hope to recover $I(t+1)$ by displacing $I(t)$ along the field $v(t+1/2)$.

If we denote

$$\mathcal{T}_{\boldsymbol{v}}(f)(\boldsymbol{x}) = f(\boldsymbol{x} - \boldsymbol{v}(\boldsymbol{x}))$$

we can expect that $I(t)$ displaced along $\boldsymbol{v}(t+1/2)$ is close to $I(t+1)$, i.e.

$$I(t+1) \simeq \mathcal{T}_{\boldsymbol{v}(t+1/2)}(I(t))$$

In general, this prediction is not perfect, so there is a prediction error defined as

$$E(t) = I(t+1) - \mathcal{T}_{\boldsymbol{v}(t+1/2)}(I(t))$$

The principle of motion compensated sequence coding is thus the following:

- We transmit the first frame $I(0)$ over the channel.
- We then transmit for each following frame the measured displacement $\boldsymbol{v}(t+1/2)$, and the prediction error $E(t)$, in order for the decoder to recover the next frame $I(t+1)$ as

$$I(t+1) = \mathcal{T}_{\boldsymbol{v}(t+1/2)}(I(t)) + E(t)$$

This is a bet that coding $\boldsymbol{v}(t+1/2)$ together with $E(t)$ is less expensive than coding $I(t+1) - I(t)$. This is conceivable, because ideally $E(t) = 0$, and the displacement field is smoother that the picture gradient $I(t+1) - I(t)$, and should be less expensive to code. The prediction error is displayed (without and with motion compensation) on Fig. 6.6 for two different image sequences.

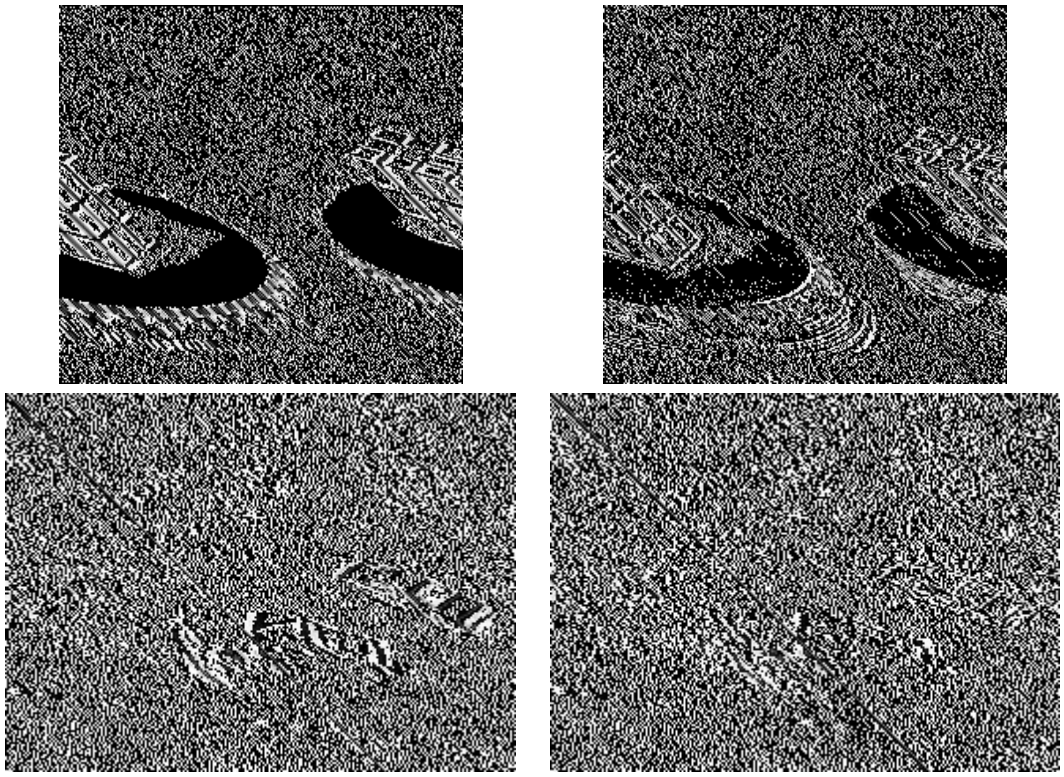
This principle has been implemented by two groups of students on the basis of the motion estimation algorithm described in the previous chapters, first by Paul Froment and Rodolphe Gintz [FG99], and then by Hervé Delay and Adrien Veillard [DV99]. In either case, a complete implementation of a coder and a decoder has been done, that compares well with available coders available freely on the internet.

6.3 Using illumination changes as an additional prediction parameter

As the illumination of a scene changes, the image modifications cannot only be described in terms of motion. Illumination changes therefore appear in the prediction error picture $E(t)$. If the illumination change profile $g(t, \boldsymbol{x})$ is smooth (as in (6.3)), the error picture is of the form

$$I(t) \times g(t) \tag{6.4}$$

which means that it inherits from the original image complexity. If on the other hand, we add a new prediction variable "illumination change" to our model, we then only have to code the illumination profile $g(\boldsymbol{x})$ we have measured, and the term (6.4) is then removed from the prediction error.



$I(t+1) - I(t)$ $I(t+1) - \mathcal{T}_{\boldsymbol{v}(t+1/2)} I(t)$

Figure 6.6: For the rubic (top) and taxi sequence (bottom), the prediction error are displayed without motion compensation (left) and with motion compensation (right). The contrast has be increased (with a saturation of the pictures on the left) so that the prediction error can be visually compared.

Thus, by adding a new illumination change variable in the compensation scheme, we replace the coding of $I(t) \times g(t)$ by the sole coding of $g(t)$ which we expect to be less expensive.

The new motion and illumination change prediction model is thus the following. We measure the displacement $\boldsymbol{v}(\boldsymbol{x}, t+1/2)$ and an illumination change field $\lambda(\boldsymbol{x}, t+1/2)$. We then predict frame $I(t+1)$ from from $I(t)$ by

$$I(t+1) = (1+\lambda(t)) \times \mathcal{T}_{\boldsymbol{v}(t+1/2)} I(t)$$

and the sequence coding scheme based on motion and illumination compensation is thus the following:

- We transmit the first frame $I(0)$ over the channel
- We transmit for each following picture $\boldsymbol{v}(t+1/2)$, $\lambda(t+1/2)$ and $E(t)$. On the basis of this information, the decoder can recover the frame $I(t+1)$ from $I(t)$ with the following formula:

$$I(t+1) = (1+\lambda(t)) \times \mathcal{T}_{\boldsymbol{v}(t+1/2)}(I(t)) + E(t)$$

Remark

In theory, illumination compensation should not be done with the formula

$$I \leftarrow I \times (1+\lambda)$$

but with the more accurate formula

$$I \leftarrow I \times \left(\frac{1+\lambda/2}{1-\lambda/2} \right)$$

because the illumination changes are supposed to be measured with respect to a reference frame at time $t+1/2$ (see Sec. 6.1.4).

6.4 Other local optic flow models

A possible extension of the presented algorithm consists in using a less simple local optic flow model than that of an optic flow that is supposed to be locally constant. Two models can also be motivated by geometric considerations [Kan89] of underlying 3–dimensional models, while staying in our fast computational framework.

6.4.1 Model of a stereographic projection of planar patches

Let us consider a stereographic projection camera model. A point of coordinates (x, y, z) is projected to a point of coordinates (X, Y) on the film plane according to the formula

$$X = \frac{x}{z}$$
$$Y = \frac{y}{z}$$

Let us now consider a planar patch in the scene. It is supported on a plane that can be characterized by its Cartesian equation. Assuming that this patch is not looked at from the edge, its Cartesian equation can be written:

$$ax + by + cz = 1 \qquad (\Pi)$$

For points on this plane, the stereographic projection can be inverted. When supposing that (x, y, z) is in the plane (Π), we obtain x, y and z from X and Y as

$$\begin{aligned} x &= \frac{X}{aX + bY + c} \\ y &= \frac{Y}{aX + bY + c} \\ z &= \frac{1}{aX + bY + c} \end{aligned}$$

If we now consider that the planar patch has a solid motion, the 3D velocity vector at a real point (x, y, z) is written as $\boldsymbol{v} = \boldsymbol{v}_0 + \boldsymbol{\omega} \times \boldsymbol{OM}$, or

$$\begin{aligned} \dot{x} &= \dot{x}_0 + \omega_y z - \omega_z y \\ \dot{y} &= \dot{y}_0 + \omega_z x - \omega_x z \\ \dot{z} &= \dot{z}_0 + \omega_x y - \omega_y x \end{aligned}$$

We can infer the optic flow from the real motion

$$\begin{aligned} \dot{X} &= \frac{\dot{x}}{z} - \dot{z}\frac{x}{z^2} \\ \dot{Y} &= \frac{\dot{y}}{z} - \dot{z}\frac{y}{z^2} \end{aligned}$$

and with the solid motion hypothesis, we get

$$\begin{aligned} \dot{X} &= \frac{\dot{x}_0 + \omega_y z - \omega_z y}{z} - x\frac{\dot{z}_0 + \omega_x y - \omega_y x}{z^2} \\ &= \frac{x_0}{z} + \omega_y - \omega_z Y - \dot{z}_0\frac{X}{z} - \omega_x XY + \omega_y X^2 \end{aligned}$$

and from the patch planarity:

$$\begin{aligned} \dot{X} &= x_0(aX + bY + c) + \omega_y - \omega_z Y - \dot{z}_0 X(aX + bY + c) - \omega_x XY + \omega_y X^2 \\ &= X^2(\omega_y - a\dot{z}_0) + XY(-\omega_x - b\dot{z}_0) \\ &\quad + X(a\dot{x}_0 - c\dot{z}_0) + Y(b\dot{x}_0 - \omega_z) + c\dot{x}_0 + \omega_y \end{aligned}$$

With similar calculus, we get:

$$\begin{aligned} \dot{Y} &= XY(-a\dot{z}_0 + \omega_y) + Y^2(-b\dot{z}_0 - \omega_x) \\ &\quad + X(a\dot{y}_0 + \omega_y) + Y(-b\dot{y}_0 - c\dot{z}_0) + c\dot{y}_0 - \omega_x \end{aligned}$$

6.4.2 Case of an orthogonal projection camera

In the case of an orthogonal projection (where X and Y are small), we obtain the same formulae without the second order terms

$$\dot{X} = X(a\dot{x}_0 - c\dot{z}_0) + Y(b\dot{x}_0 - \omega_z) + c\dot{x}_0 + \omega_y$$
$$\dot{Y} = X(a\dot{y}_0 + \omega_y) + Y(-b\dot{y}_0 - c\dot{z}_0) + c\dot{y}_0 - \omega_x$$

6.4.3 Estimation of a non locally constant flow with wavelets

In this prospective section, we only consider the simpler case of the orthogonal projection camera model, although everything below can be extended to the more general case of the stereographic projection.

The flow model is then of the form

$$v_x = ax + by + c$$
$$v_y = dx + ey + f$$

where we have to identify the coefficients a, b, c, d, e and f. The assumption according to which the flow is constant over the support of the considered wavelets is replaced here with the assumption that it is locally linear. The projection of the optic flow equation on the wavelets is then

$$\left\langle \frac{\partial I}{\partial x} v_x, \psi^n \right\rangle + \left\langle \frac{\partial I}{\partial y} v_y, \psi^n \right\rangle + \left\langle \frac{\partial I}{\partial t}, \psi^n \right\rangle = 0$$

and by application of the flow model

$$\left\langle \frac{\partial I}{\partial x}(ax + by + c), \psi^n \right\rangle + \left\langle \frac{\partial I}{\partial y}(dx + ey + f), \psi^n \right\rangle + \left\langle \frac{\partial I}{\partial t}, \psi^n \right\rangle = 0$$

and after an integration by parts

$$a\left(\left\langle xI, \frac{\partial \psi^n}{\partial x} \right\rangle + \langle I, \psi^n \rangle\right) + b\left\langle yI, \frac{\partial \psi^n}{\partial y} \right\rangle + c\left\langle I, \frac{\partial \psi^n}{\partial x} \right\rangle$$
$$+ d\left\langle xI, \frac{\partial \psi^n}{\partial x} \right\rangle + e\left(\left\langle yI, \frac{\partial \psi^n}{\partial y} \right\rangle + \langle I, \psi^n \rangle\right) + f\left\langle I, \frac{\partial \psi^n}{\partial y} \right\rangle = \left\langle \frac{\partial I}{\partial t}, \psi^n \right\rangle$$

Thus, the coefficients a to f are solution of N linear equations whose coefficients are wavelet coefficients of I, xI and yI. We can again estimate these coefficients in a short time ($\mathcal{O}(N)$), the same way as for the simpler model.

Remark

The reader can figure out that for the more general model of a locally quadratic flow, a similar result holds, that the coefficients of the linear systems we obtain are wavelet coefficients of the pictures I, xI, yI, x^2I, y^2I and xyI.

Such richer models have several uses, and advantages over the simpler model of locally constant optic flow. As has already been introduced in Sec. 4.1.3, such a model can be used for motion segmentation, and Odobez and Bouthemy demonstrated this in [OB98]. A simpler model detects discontinuities of the flow where the flow variations are fast, like in the middle of a planar patch that has a zoom motion w.r.t the camera, and is thus inappropriate.

Such models can also make the flow estimation more robust as shown by Mendelsohn *et al.* [MSB97], who use a locally planar object model with a stereographic projection. Their measurement of the flow is enhanced, especially for large displacements. Another possible advantage would be to enhance three dimensional scene reconstruction by imposing a locally smooth object model.

Bibliography

[BF93] P. Bouthemy and E. François. Motion segmentation and qualitative dynamic scene analysis from an image sequence. *Int. Journal of Computer Vision*, 10(2):157–182, 1993.

[BFB94] J. Barron, D. Fleet and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[DV99] H. Delay and A. Veillard. Compression de séquences d'images par une méthode de flot. Technical report, École Polytechnique, France, École Polytechnique Fédérale de Lausanne, March 1999.

[FG99] P. Froment and R. Gintz. Mesure du flot optique et compression vidéo. Technical report, Études d'Approfondissement, École Polytechnique, France, January 1999.

[FJ90] D. Fleet and A. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5:77–104, 1990.

[Kan89] K. Kanatani. *Group-theoretical methods in image understanding*. Springer Series in Information Sciences. Springer, 1989.

[MK98] J. Magarey and N. Kingsbury. Motion estimation using a complex–valued wavelet transform. *IEEE Trans. on Signal Processing*, 46(4):1069–1084, April 1998.

[MP98] E. Mémin and P. Pérez. Dense estimation and object-based segmentation of the optical flow with robust techniques. *IEEE Trans. Image Processing*, 7(5):703–719, May 1998.

[MSB97] J. Mendelsohn, E. Simoncelli and R. Bajcsy. Discrete–time rigidity–constrained optical flow. In *7th Int'l Conf Computer Analysis of Images and Patterns, Kiel, Germany*. September 1997.

[OB98] J.-M. Odobez and P. Bouthemy. Direct incremental model-based image motion segmentation for video analysis. *Signal Processing*, 6(2):143–155, 1998.

[Sim98] E. P. Simoncelli. Bayesian multi–scale differential optical flow. In H. Jähne and Geissler, editors, *Handbook of computer vision and applications*. Academic Press, 1998.

[WKCL98] Y.-T. Wu, T. Kanade, J. Cohn and C.-C. Li. Optical flow estimation using wavelet motion model. In *Sixth International Conference on Computer Vision*, pages 992–998. Narosa Publishing House, 1998.

[WM95] J. Weber and J. Malik. Robust computation of optical flow in a multi-scale differential framework. *International Journal of Computer Vision*, 14(1):5–19, 1995.

Appendix A

Derivative wavelet bases and frames

Abstract

In some applications, like in our case for the optic flow estimation, we use derivative wavelets of a basis to compute the coefficients of a signal derivative. We will see that the differentiation can easily be done in the framework of multiresolution analyses. If a wavelet arises in a multiresolution analysis with finite impulse response filters, the same holds for the differentiated wavelet under weak assumptions. We also see that the derivatives of a wavelet frame make up a frame, up to a renormalization.

A.1 Wavelet and filter derivatives

Let ϕ and ψ be a scaling function and wavelet built from two finite impulse response filters m_0 and m_1 with infinite convolutions:

$$\hat{\phi}(\omega) = \prod_{k=1}^{+\infty} m_0\left(\frac{\omega}{2^k}\right) \tag{A.1a}$$

$$\hat{\psi}(\omega) = m_1\left(\frac{\omega}{2}\right)\hat{\phi}\left(\frac{\omega}{2}\right) \tag{A.1b}$$

We call $\tilde{m}_0$ and $\tilde{m}_1$ the dual filters and $\tilde{\phi}$ and $\tilde{\psi}$ the dual filters defined from $\tilde{m}_0$ and $\tilde{m}_1$ with formulae similar to (A.1a,b).

The Fourier transforms of the derivatives of ϕ and ψ are respectively $\omega \mapsto i\omega\hat{\phi}(\omega)$ and $\omega \mapsto i\omega\hat{\psi}(\omega)$. To obtain the corresponding discrete filters, we use the following algebraic

identity:

$$\begin{aligned} e^{i\omega} - 1 &= (e^{i\omega/2} + 1)(e^{i\omega/2} - 1) \\ &= (e^{i\omega/2} + 1)(e^{i\omega/4} + 1)(e^{i\omega/4} - 1) \\ &\vdots \\ &= \prod_{k=1}^{K} (e^{i\omega/2^k} + 1) \times (e^{i\omega/2^K} - 1) \\ &= \prod_{k=1}^{K} \left(\frac{e^{i\omega/2^k} + 1}{2} \right) \times 2^K (e^{i\omega/2^K} - 1) \end{aligned}$$

and by taking the limit:

$$e^{i\omega} - 1 = \prod_{k=1}^{+\infty} \left(\frac{e^{i\omega/2^k} + 1}{2} \right) \times i\omega$$

thus

$$i\omega = \frac{e^{i\omega} - 1}{\prod_{k=1}^{+\infty} \left(\frac{e^{i\omega}+1}{2} \right)}$$

By setting

$$\begin{aligned} M_0(\omega) &= \frac{2m_0(\omega)}{e^{i\omega} + 1} \\ M_1(\omega) &= \frac{e^{i\omega} - 1}{2} m_1(\omega) \end{aligned}$$

the new wavelets Φ and Ψ built by infinite convolution of the filters M_0 and M_1 (with formulae similar to A.1a–A.1b) have the following properties:

$$\phi'(t) = \Phi(t+1) - \Phi(t) \tag{A.2a}$$

$$\psi'(t) = \Psi(t) \tag{A.2b}$$

The dual filters are obtained with the reverse transformation. If we set

$$\begin{aligned} \tilde{M}_0(\omega) &= \frac{e^{-i\omega} + 1}{2} \tilde{m}_0(\omega) \\ \tilde{M}_1(\omega) &= \frac{2\tilde{m}_1(\omega)}{e^{-i\omega} - 1} \end{aligned}$$

the dual wavelets $\tilde{\Phi}$ and $\tilde{\Psi}$, if they exist, fulfill the following equations:

$$\tilde{\phi}(t) = \int_{t-1}^{t} \tilde{\Phi}(u) du \tag{A.3a}$$

$$\tilde{\psi}(t) = \int_{t}^{+\infty} \tilde{\Psi}(u) du \tag{A.3b}$$

We see that systematically, if the original transfer matrix is invertible, the derivative filter transfer matrix also is, since

$$\begin{vmatrix} m_0(\omega) & m_0(\omega+\pi) \\ m_1(\omega) & m_1(\omega+\pi) \end{vmatrix} = - \begin{vmatrix} M_0(\omega) & M_0(\omega+\pi) \\ M_1(\omega) & M_1(\omega+\pi) \end{vmatrix}$$

However (if for example the functions are not smooth enough, like Haar wavelets), nothing guarantees that the wavelet families we obtain still are unconditional bases of $L_2(\mathbb{R})$.

To stay in the framework of biorthogonal multiresolution analyses in $L_2(\mathbb{R})$, the derivative wavelets also have to be square integrable. The translated scaling functions also need to make up a Riesz basis of the corresponding scale space. Cohen, Daubechies and Feauveau proved a theorem that can be used to establish that compactly supported filters define through formulae (A.1a–A.1b) biorthogonal unconditional bases of $L_2(\mathbb{R})$.

This theorem imposes a minimum regularity constraint on the scaling functions. Let us remind it.

Theorem A.1 (Cohen, Daubechies, Feauveau)

Let (m_0, m_1) and $(\tilde{m}_0, \tilde{m}_1)$ be two pairs of filters fulfilling the following duality relation

$$\begin{bmatrix} m_0(\omega) & m_0(\omega+\pi) \\ m_1(\omega) & m_1(\omega+\pi) \end{bmatrix} \times \begin{bmatrix} \tilde{m}_0(\omega) & \tilde{m}_0(\omega+\pi) \\ \tilde{m}_1(\omega) & \tilde{m}_1(\omega+\pi) \end{bmatrix}^T = I \tag{A.4}$$

We assume that m_0 and $\tilde{m}_0$ can be factored as

$$m_0(\omega) = \left(\frac{e^{i\omega}+1}{2}\right)^N r(\omega)$$
$$\tilde{m}_0(\omega) = \left(\frac{e^{i\omega}+1}{2}\right)^{\tilde{N}} \tilde{r}(\omega)$$

where the exponents N and $\tilde{N}$ are of highest degree possible and r and $\tilde{r}$ are polynomials. We define the critical exponents

$$b = \inf_{j\geq 1}\left[\frac{1}{j\log 2}\log \max_{\omega\in\mathbb{R}}\left|\prod_{k=1}^{j} r(2^k\omega)\right|\right]$$
$$\tilde{b} = \inf_{j\geq 1}\left[\frac{1}{j\log 2}\log \max_{\omega\in\mathbb{R}}\left|\prod_{k=1}^{j} \tilde{r}(2^k\omega)\right|\right]$$

If $N - b > 1/2$ and $\tilde{N} - \tilde{b} > 1/2$, the wavelet bases built with the pairs (ϕ,ψ) and (($\tilde{\phi}$,$\tilde{\Psi}$) are biorthogonal unconditional wavelet bases of $L_2(\mathbb{R})$.

This theorem proves that if the scaling function (of compact support) are smooth enough, then the two dual wavelet families obtained this way are unconditional bases.

If we apply this theorem to the new filter pairs (M_0, M_1) and $(\tilde{M}_0, \tilde{M}_1)$, the required assumption on the original filters can easily be derived:

$$N - b > 3/2$$

This condition consists in requiring from ϕ a stronger smoothness (which was predictable since now ϕ' has to be in $L_2(\mathbb{R})$. Thus the following theorem, which is a consequence of Th. A.1.

Theorem A.2

Let (m_0, m_1) and $(\tilde{m}_0, \tilde{m}_1)$ be two dual filter pairs. let N, $\tilde{N}$, b and $\tilde{b}$ be defined as in Th. A.1. If $N - b > 3/2$ and $\tilde{N} - \tilde{b} > 1/2$, then the family $(2^{-j}\psi'_{jk})_{j,k\in\mathbb{Z}}$ is an unconditional basis of $L_2(\mathbb{R})$. The associated dual basis is the family of functions

$$x \mapsto \int_x^{+\infty} 2^j \tilde{\psi}_{jk}(u)du$$

Proof. The proof consists in showing that the two pairs (M_0,M_1) and $(\tilde{M}_0, \tilde{M}_1)$ fulfill the requirements of Th. A.1. The filters M_0 and $\tilde{M}_0$ can be factored in a way similar to m_0 and $\tilde{m}_0$:

$$M_0(\omega) = \left(\frac{e^{i\omega}+1}{2}\right)^{N-1} r(\omega)$$
$$\tilde{M}_0(\omega) = \left(\frac{e^{i\omega}+1}{2}\right)^{\tilde{N}+1} \tilde{r}(\omega)$$

while the critical exponents of M_0 and $\tilde{M}_0$ are respectively the same as those of m_0 and $\tilde{m}_0$.

Finally, the filters M_1 and $\tilde{M}_1$ are compactly supported. For M_1, this is easy to derive since M_1 is the convolution of m_1 with another finite impulse response filter. $\tilde{M}_1$ however can be written as

$$\tilde{M}_1(\omega) = \frac{2\tilde{m}_1(\omega)}{e^{-i\omega}-1} \tag{A.5}$$

and can be polynomial only if $\tilde{m}_1$ has at least one vanishing moment (i.e. $\tilde{m}_1(0) = 0$). Let us consider for this the equation (A.4) for $\omega = 0$. Since b is necessarily positive, N is not less than 2. This implies that $m_0(\pi) = 0$. The transfer matrix $T(0)$ is upper triangular. Since $\tilde{T}(0) = T(0)^{-T}$, $\tilde{T}(0)$ is lower triangular which means that $\tilde{m}_1(0) = 0$ and the trigonometric polynomial is thus a multiple of $e^{i\omega} - 1$.

This allow to say in conclusion that Th. A.1 applies to the wavelet family $(2^j\psi'_{jk})$. Equations (A.2a–A.2b) and (A.3a–A.3b) are direct consequences of the definition of the filters M_0, M_1, $\tilde{M}_0$ and $\tilde{M}_1$. □

Remark

The definition of derivative filters can be extended for multiresolution that are a little more general than the above ones: non stationary multiresolutions. If a wavelet as an infinite convolution of discrete filters:

$$\hat{\psi}(\omega) = \prod_{j=1}^{+\infty} m_j(2^{-j}\omega)$$

then, the derivative can be written

$$\widehat{\psi'}(\omega) = \prod_{j=1}^{+\infty} M_j(2^{-j}\omega)$$

where

$$M_j(\omega) = \begin{cases} \frac{e^{i\omega}-1}{2} m_j(\omega) & \text{if } j = 0 \\ \frac{2m_j(\omega)}{e^{i\omega}+1} & \text{if } j > 0 \end{cases}$$

We must notice that using such wavelets only makes sense if the m_j filters are almost all the same, up to the very first ones, or else we loose the fast computational structure of the dyadic filter banks.

A.2 Gradient wavelet frames

In this section, we establish a result that states that under some weak conditions, the renormalized derivatives of a wavelet frame make up a frame.

For this, we first prove the following lemma on dual frames.

Lemma A.1 (Scissor lemma.)

Let $(e_n)_{n\in\mathbb{N}}$ and $(f_n)_{n\in\mathbb{N}}$ be families of vectors of a Hilbert space H, that fulfill the following upper bound inequality:

$$\sum_{n\in\mathbb{N}} |\langle u, e_n\rangle|^2 \leq M_e \|u\|^2 \qquad \text{for all } u \in H \tag{A.6a}$$

$$\sum_{n\in\mathbb{N}} |\langle u, f_n\rangle|^2 \leq M_f \|u\|^2 \qquad \text{for all } u \in H \tag{A.6b}$$

and also fulfill

$$\langle u, u\rangle = \sum_{n\in\mathbb{N}} \langle u, e_n\rangle\langle f_n, u\rangle \qquad \text{for all } u \in H \tag{A.7}$$

then $(e_n)_{n\in\mathbb{N}}$ and $(f_n)_{n\in\mathbb{N}}$ are dual frames, which means that

$$\frac{1}{M_f}\|u\|^2 \leq \sum |\langle u, e_n\rangle|^2 \qquad \forall u \in H \tag{A.8a}$$

$$\frac{1}{M_e}\|u\|^2 \leq \sum |\langle u, f_n\rangle|^2 \qquad \forall u \in H \tag{A.8b}$$

$$u = \sum_{n\in\mathbb{N}} \langle u, f_n\rangle e_n = \sum_{n\in\mathbb{N}} \langle u, e_n\rangle f_n \qquad \forall u \in H \tag{A.8c}$$

Proof. The proof is fairly simple, and is inspired from work by Cohen [Coh92]. Because both families fulfill the stability inequality (A.6a–A.6b) the right hand side of (A.7) is absolutely convergent (by Cauchy–Schwartz's inequality).

This same inequality also implies

$$\|u\|^2 \leq \sqrt{\sum_{n\in\mathbb{N}} |\langle u, e_n\rangle|^2}\sqrt{\sum_{n\in\mathbb{N}} |\langle u, f_n\rangle|^2}$$

By comparing this with (A.6a) and (A.6b), we respectively get the lower bound inequalities (A.8b) and (A.8a). Both families are therefore frames. By polarization of (A.7), we obtain

$$\langle u, v\rangle = \sum_{n\in\mathbb{N}} \langle u, e_n\rangle\langle f_n, v\rangle \qquad \text{for all } u, v \in H$$

and (A.8c). □

This lemma means that if two vector families fulfill the upper frame inequality and the relation (A.7), then they also fulfill the lower frame inequality and are dual frames. Here, two frames are said to be dual if they fulfill (A.8c). With such a definition, a frame may have several dual frames. In the wavelet literature, there is also a definition for *the* dual frame of a given frame, which is then the frame that provides minimum ℓ^2 norm expansion coefficients. We do not need to know exactly what the dual frame of a frame is to use the scissor lemma. We will use it to prove that the gradient of a frame of wavelets also makes up a frame (up to some renormalization, of course).

We can now state a theorem of wavelet derivative frames:

Theorem A.3

Assume $\{\psi^n_{j\boldsymbol{k}} : n = 1\dots N, j \in \mathbb{Z}, \boldsymbol{k} \in \mathbb{Z}^2\}$ *where* $\psi^n_{j\boldsymbol{k}}(\boldsymbol{x}) = 2^j\psi^n(2^j\boldsymbol{x} - \boldsymbol{k})$ *is a frame of* $L_2(\mathbb{R})$. *Denote* $\tilde{\psi}^n_{j\boldsymbol{k}}$ *the wavelets of the dual frame. Assume:*

$$|\widehat{\psi^n}(\boldsymbol{\omega})| \leq \frac{C}{(1+|\omega_x|)^s(1+|\omega_y|)^s} \qquad \text{for some } s > 3/2 \qquad \text{(A.9a)}$$

$$|\widehat{\tilde{\psi}^n}(\boldsymbol{\omega})| \leq \frac{C}{(1+|\omega_x|)^s(1+|\omega_y|)^s} \qquad \text{for some } s > 1/2 \qquad \text{(A.9b)}$$

$$|\widehat{\psi^n}(\boldsymbol{\omega})| \leq C'|\boldsymbol{\omega}|^\alpha \qquad \text{for some } \alpha > 0 \qquad \text{(A.9c)}$$

$$|\widehat{\tilde{\psi}^n}(\boldsymbol{\omega})| \leq C'|\boldsymbol{\omega}|^\alpha \qquad \text{for some } \alpha > 1 \qquad \text{(A.9d)}$$

Then, the wavelet family

$$\left\{2^{-j}\frac{\partial\psi^n_{j\boldsymbol{k}}}{\partial x_\ell} : j \in \mathbb{Z}, \boldsymbol{k} \in \mathbb{Z}^2, n \in [1, N], \ell \in [1, 2]\right\} \qquad \text{(A.10)}$$

is a frame of $L_2(\mathbb{R}^2)$.

Proof. We now exhibit a dual frame. If we denote

$$\Psi^{n,\ell}_{j\boldsymbol{k}} = 2^{-j}\frac{\partial\psi^n_{j\boldsymbol{k}}}{\partial x_\ell}$$

then we set

$$\tilde{\Psi}^{n,\ell}_{j\boldsymbol{k}}(x_1, x_2) = 2^j G_\ell * \tilde{\psi}^n_{j\boldsymbol{k}}$$

where the distribution G_ℓ is defined by its Fourier transform

$$\hat{G}_\ell(\omega_1, \omega_2) = \frac{\omega_\ell}{\omega_1^2 + \omega_2^2}$$

In fact, $\boldsymbol{G} = [G_1\, G_2]^T$ is the gradient of Green's kernel for the heat equation:

$$\Delta u = v$$

The assumption (A.9d) guarantees that $\tilde{\Psi}^{n,\ell}_{j\boldsymbol{k}}$ are in $L_2(\mathbb{R}^2)$. Moreover, we can write

$$\iint f(\boldsymbol{x}) g(\boldsymbol{x}) d\boldsymbol{x} = \iint \boldsymbol{\nabla} f \cdot (\boldsymbol{G} * g) d\boldsymbol{x} \tag{A.11}$$

for functions f and g smooth enough and of zero integral, which is obvious if we look at the Fourier transforms.

Assumptions (A.9a) and (A.9d) are tailored so that wavelets $\Psi^{n\ell}$ and $\tilde{\Psi}^{n,\ell}$ fulfill the assumptions of lemma 5.1. This provides us with the inequalities:

$$\sum_{j,\boldsymbol{k},n,\ell} |\langle f, \Psi^{n\ell}_{j\boldsymbol{k}}\rangle|^2 \leq M\|f\|^2 \qquad \forall f \in L_2(\mathbb{R}^2)$$

$$\sum_{j,\boldsymbol{k},n,\ell} |\langle f, \tilde{\Psi}^{n\ell}_{j\boldsymbol{k}}\rangle|^2 \leq \tilde{M}\|f\|^2 \qquad \forall f \in L_2(\mathbb{R}^2)$$

We denote by $\hat{C}_0^\infty$ the set of functions whose Fourier transform is C^∞ and of compact support in $\mathbb{R}^2 - \{0\}$. For all $f \in \hat{C}_0^\infty$, we can write by duality of $(\psi^n_{j\boldsymbol{k}})$ and $(\tilde{\psi}^n_{j\boldsymbol{k}})$

$$\sum_{\ell=1,2} \left\langle \frac{\partial f}{\partial x_\ell}, G_\ell * f \right\rangle = \sum_{j,\boldsymbol{k},n,\ell} \left\langle \frac{\partial f}{\partial x_\ell}, \psi^n_{j\boldsymbol{k}} \right\rangle \left\langle \tilde{\psi}^n_{j\boldsymbol{k}}, G_\ell * f \right\rangle$$

By application of (A.11), we get

$$\langle f, f\rangle = \sum_{j,\boldsymbol{k},n,\ell} \left\langle \frac{\partial f}{\partial x_\ell}, \psi^n_{j\boldsymbol{k}} \right\rangle \left\langle \tilde{\psi}^n_{j\boldsymbol{k}}, G_\ell * f \right\rangle$$

and because the operators $\frac{\partial}{\partial x_\ell}$ and G_ℓ are skew symmetric, we obtain

$$\langle f, f\rangle = \sum_{j,\boldsymbol{k},n,\ell} \left\langle f, \frac{\partial \psi^n_{j\boldsymbol{k}}}{\partial x_\ell} \right\rangle \left\langle G_\ell * \tilde{\psi}^n_{j\boldsymbol{k}}, f \right\rangle$$

and thus

$$\langle f, f\rangle = \sum_{j,\boldsymbol{k},n,\ell} \langle f, \Psi^{n\ell}_{j\boldsymbol{k}}\rangle \left\langle \tilde{\Psi}^{n\ell}_{j\boldsymbol{k}}, f \right\rangle$$

By density of $\hat{C}_0^\infty$ in $L_2(\mathbb{R}^2)$, and because the right hand side of the above expression is continuous in $L_2(\mathbb{R}^2)$, this equality holds for any $f \in L_2(\mathbb{R}^2)$ and applying the scissor lemma finishes the proof. □

Bibliography

[BCD+91] G. Beylkin, R. Coifman, I. Daubechies, S. G. Mallat, Y. Meyer, L. Raphael and M. B. Ruskai. *Wavelets*. Jones and Bartlett, Boston, 1991.

[CDF92] A. Cohen, I. Daubechies and J. Feauveau. Bi-orthogonal bases of compactly supported wavelets. *Comm. Pure and Appl. Math.*, 45:485–560, 1992.

[Coh92] A. Cohen. *Ondelettes et traitement numérique du signal*. Masson, 1992.

[Dau92] I. Daubechies. *Ten Lectures on Wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[Mal89a] S. G. Mallat. Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$. *Trans. Amer. Math. Soc.*, 315(1):69–87, 1989.

[Mal89b] S. G. Mallat. A theory for multiscale signal decomposition: The wavelet representation. *IEEE Trans. on Pattern and Machine Intelligence*, 11(7):674–693, 1989.

[Mal97] S. G. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1997.

Part II

Learning and interpolation

Chapter 7

Learning

Abstract

Learning and interpolation are two approaches to solve the problem of how to build a reasonable estimate of an unknown function on the basis of a finite number of samples. Such problems arise in various frameworks ranging from partial differential equations through geometric modeling in image synthesis to learning and adaptive control. In this chapter, we present an overview of various existing methods the purpose of which is to estimate functions from samples.

7.1 The learning problem

The classical problem that each method has to solve can be stated in the following way: given a number of measures $(x_n, y_n) \in \mathbb{R}^d \times \mathbb{R}$, for $n = 1 \dots N$, we want to find a function f mapping $\mathbb{R}^d$ to $\mathbb{R}$ such that

$$f(x_n) = y_n \qquad \text{for } n = 1 \dots N$$

7.1.1 What is the best solution?

Such a problem can easily be solved in the framework of parametric estimation, where the unknown function is determined by a small number of parameters (as in linear regression). In such a case, the underlying linear system is overdetermined and has usually no solution. A model of measures with a Gaussian noise can be used:

$$y_n = f(x_n) + \epsilon_n$$

where the ϵ_n are i.i.d. Gaussian variables of zero mean and of standard deviation σ. Regression consists in finding the parameter combination that maximizes the likelihood (conditional density w.r.t the function f). In the Gaussian case, we maximize

$$L = \frac{1}{(8\pi)^{N/2}\sigma^N} \prod_{n=1}^{N} e^{-(f(x_n)-y_n)^2/2\sigma^2}$$

which is the same as minimizing

$$\sum_{n=1}^{N}(f(x_n)-y_n)^2$$

and we obtain what is usually called a least square solution.

If no such strong assumption is made on the unknown function f, and we suppose it lives in some infinite dimensional vector space, we leave the framework of parametric estimation and the problem is ill-posed. We have to use additional constraints in order for the function f to predict correctly new samples.

7.1.2 What is the best problem?

A quite natural heuristic consists in saying that the "simplest" interpolant is the best. This notion has been transcribed in mathematical terms in several ways:

- an analytic way: a function is said to be simple if it is smooth. This motivates approaches based on regularization [TA77], that can be seen as a Bayesian estimation.
- an information theoretical way : a function is simple if it can be described with a small number of words (or a short program). This has motivated work based on complexity as defined by Kolmogorov.
- a statistical way: Vapnik [Vap95, Vap97] has built a statistical theory of learning. In his framework, the purpose is to estimate an output random variable y with a function $f(x)$ of an input random variable, on the basis of samples of x and y. The prediction risk $E(\tilde{f}(x)-y)^2$ cannot be estimated. Vapnik's theory however provides a bound on this risk which is the sum of two terms, the empirical risk (average error on the past samples), and a complexity term that depends on some complexity measure (Vapnik-Chervonenkis dimension) of the function subfamily in which we choose an interpolant. The formula is

  $$E((y-f(x))^2 \leq \frac{1}{N}\sum_{n=1}^{N}(y_n-f(x_n))^2 + H(h,\eta,N)$$

  with probability larger that $1-\eta$. h is the Vapnik-Chervonenkis dimension. Vapnik suggests to minimize what he calls the structural risk, which is the sum of the empirical risk and the complexity term.

These approaches can be motivated by a number of criteria, that are:

- convergence and accuracy criteria: ability to estimate accurately function from a given smoothness class
- a practical criterion of computation time. Most learning problems involve numerous data of very high dimension. Computational efficiency most of the time means feasibility.

- storage constraints. They are motivated by themselves as well as through their influence on the computational complexity. They are also motivated by the heuristic that the simplest solution is also likely to be the best.

7.1.3 Error estimation and target functional

We can define several types of estimation error. The simplest is the quadratic empirical error:

$$R_{\text{emp}} = \frac{1}{N}\sum_{n=1}^{N}(y_n - f(x_n))^2$$

In various approaches to learning, one wants to control the prediction risk. We suppose that the measures x_n and y_n are realizations of random variables X and Y that are correlated, of joint probability measure $p(x, y)dxdy$. The risk is then

$$R = E((Y - f(X))^2) = \iint (y - f(x)^2 p(x, y)dxdy$$

This risk cannot be estimated. Some approximation techniques then consist in minimizing in replacement a target functional that can be measured, and most suggested error metrics are composite error functionals, that is the sum of two terms:

- a matching functions (or empirical error)
- a regularity or complexity term, sometimes also called "capacity control term".

7.1.4 Different approaches

The available results in the various approximation methods have very different properties. For some approaches, we only get very limited results on the approximation algorithms, while other techniques have stronger theoretical foundations and are known to converge.

7.2 Neural networks

The expression "neural networks" is used for a wide variety of techniques we cannot review here. In this work, we use therefore this term to denote a *deterministic multilayer perceptron*.

7.2.1 The multilayer perceptron

The multilayer perceptron is composed of several units called *neurons*, placed on layers numbered from 0 to L. The neurons on layer 0 are called *input neurons* and their set is denoted by $\mathcal{I}$ while the neurons in layer L are called *output neurons* and their set is called $\mathcal{O}$.

The neurons are connected together by oriented connections (*axons*)that go from a neuron of a given layer ℓ to a neuron of the next layer $\ell + 1$. We denote the relation "the neuron

n is connected to the neuron n'" by "$n \to n'$". Each neuron can have a *state* $x_n \in \mathbb{R}$. Each connection going from a neuron n to neuron n' has a *weight* $c_{nn'}$.

We can *query* a neural network. This consists in setting the state of the input neurons to some input vector value, to compute the resulting state of the other neurons and to read the state of the output neurons. To compute the state of the non-input neurons, we associate to each neuron a fixed transfer function $\phi_n : \mathbb{R} \to \mathbb{R}$. The state of a neuron n is computed by

$$x_n = \phi_n \left(\sum_{n' \to n} c_{n'n} x'_n \right) \tag{7.1}$$

The sum is written over all neurons n' such that there exists a connection going from n' to n, which means that the involved neurons n' are all at the layer prior to that of n. The computation can be done with a single pass through the layers in increasing order, and thus called *forward propagation*.

The construction of a neural network consists in choosing the number of neurons and their transfer functions, and to choose the connections and fix their weights. This approach is attractive, because by simply changing the weights, we can represent a wide variety of functions (provided there are enough layers and neurons), and because there exists a generic way to tune the weights of the connections, called *supervised learning rule*. This rule can be used to make the neural network reproduce a dependence between two random variables I and O. I and O are two vectors $I = (I_n)_{n \in \mathcal{I}}$ and $O = (O_n)_{n \in \mathcal{O}}$.

7.2.2 Expression capacity

One of the main advantages of the multilayer perceptron is its ability to approximate efficiently any mapping from an input to an output, provided it has enough neurons. With a two layer perceptron with sigmoidal transfer functions, we can approximate any characteristic function of a convex set in a natural way. A less intuitive result is maybe that any smooth function may be approximated arbitrarily well by finite combinations of sigmoid functions [Cyb89]. This is the same as saying the functions

$$\boldsymbol{x} \mapsto \sum_k c_k \sigma(\boldsymbol{a}_k \cdot \boldsymbol{x} + b)$$

are dense. More detailed approximation results have been published by Barron [Bar93].

Recently, Candès and Donoho have suggested to replace systematically the sigmoids with ridgelets, i.e. functions of the form

$$\boldsymbol{x} \mapsto \psi(\boldsymbol{a} \cdot \boldsymbol{x} + b)$$

and built in $L_2(\mathbb{R}^d)$ a continuous ridgelet transform and ridgelet frames. They provide an explicit method to construct a finite ridgelet approximation of a known function.

7.2.3 Supervised learning rule

We have a source of examples: a pair of correlated random variables (I, O). We submit *examples* to the network, which means we draw samples of these variables. We query the

network by setting the input neuron states to I and we compare the output neuron states $f(I)$ to O. Ideally, the output is equal to O (or if the relationship between I and O is non deterministic, the expectancy of $f(I) - O$ is zero).

However, it is usually not the case, so the learning rule then consists in modifying the connection weights on the basis of their impact on the output with a simple stochastic gradient, which is:

$$c_{nn'} \leftarrow c_{nn'} - \gamma \frac{\partial f(I)}{\partial c_{nn'}} \cdot (f(I) - O)$$

This rule can be implemented in a very simple form called error backpropagation. We define for each neuron n a second state variable which is its error r_n. This error is defined for the output variable by $r_n = x_n - O_n$, and is then propagated back according to the following formula:

$$r_n = \sum_{n \to n'} c_{nn'} r_{n'} \phi'_{n'}(x_{n'})$$

thus the word "backpropagation". The updates of the coefficients $c_{nn'}$ are then:

$$c_{nn'} \leftarrow c_{nn'} - \gamma x_n r_{n'}$$

These two propagation and update rules are usually called the Widrow-Hoff rules.

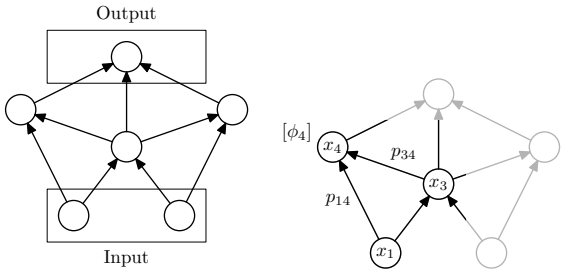


Figure 7.1: Sample (simple) neural network. On the left, we can see the neurons and their connections. The input and output neurons are marked. On the right, the propagation rule is illustrated: $x_4 = \phi_4(p_{14}x_1 + p_{34}x_3)$.

The transfer functions ϕ_n should not be all linear. Otherwise, the relationship between input and output is linear, and is then trivial. We thus define the "true" number of layers of a multilayer perceptron as the number of layers where the transfer functions are not linear.

The transfer functions are in general regular. This is required in order that close inputs are mapped to close outputs, and is supposed to be connected to the ability to "generalize" a correspondence on the basis of some examples. Widely used functions are sigmoids σ, like the hyperbolic tangent $\sigma : x \mapsto \operatorname{th} x$. Such functions go to -1 at $-\infty$ and to $+1$ at $+\infty$, and can therefore almost be considered as binary output functions.

This framework is very large, so that almost any function representation can be seen as a neural network implementation. Most notably, as soon as we have chosen a basis $(g_m)_{m\in\mathbb{N}}$ of a function space on $\mathbb{R}^d$, expanding any function over this basis consists in building a single layer neural network that implements it. The input is connected to as many neurons at layer 1 that have each a transfer function g_m, and whose outputs are combined into a single output at layer 2.

The major drawback of these approaches is that nothing guarantees that a given neural network will be able to learn a given correspondence between I and O with supervised learning, unless the weights we have to tune are located after the last nonlinear transfer function. In this special case, the error function is a quadratic (thus convex) function of the weights, and there exists a unique minimum which is global. Otherwise, the error function $\mathrm{E}\left(\|f(I)-O\|^2\right)$ may be a highly non convex function of the weights $c_{nn'}$, and a learning algorithm may be stuck in a local minimum.

7.2.4 The three steps of neural network design

In practice however, the difficult parts of this approach are hidden in the steps prior to the tuning of connection weights. To summarize, we can consider that the construction of a neural network can be split into three steps:

- The choice of an architecture. This consists in choosing the number of layers, and to choose some connection weights and the possible transfer functions.
- The choice of a network growth rule. In practice, the number of neurons to use to represent a given mapping from inputs to outputs depends on the complexity of this mapping. We thus have to set up rules that determine how the structure of the network is supposed to change w.r.t the mapping to learn. Such rules can also include neuron suppression rules.
- The choice of a learning rule to estimate the remaining weights. The most universal rule is the supervised learning rule described above. Other rules can be used however, if the structure of the network allows it. Usually, the feasibility and the convergence of this step depends crucially on the two first steps.

7.2.5 Rosenblatt's perceptron

The simplest form of neural network, a single layer neural network has been described by Rosenblatt in 1962. Rosenblatt named it the *perceptron*.

The perceptron has N scalar inputs $x = (x_1, \ldots, x_N)$ and a single binary output y:

$$y = \mathrm{sgn}(c + w \cdot x)$$

where c is a scalar and v a N dimensional vector. The perceptron thus splits the space into two half spaces of equations $w \cdot x + c < 0$ and $w \cdot x + c > 0$.

Rosenblatt wanted to "teach" the perceptron how to distinguish two classes of samples $x \in \mathcal{C}^+$ and $x \in \mathcal{C}^-$ that are supposedly separable with a hyperplane, i.e. there exists w and

c such that

$$\begin{aligned} w \cdot x + c &< 0 && \forall x \in \mathcal{C}^- \\ w \cdot x + c &> 0 && \forall x \in \mathcal{C}^+ \end{aligned}$$

On the basis of biologic considerations, he suggests to draw a set of samples (x_t, ϵ_t) where

$$\begin{aligned} \epsilon_t &= +1 && \text{if } x_t \in \mathcal{C}^+ \\ \epsilon_t &= -1 && \text{if } x_t \in \mathcal{C}^- \end{aligned}$$

and to modify recursively the coefficients w and c each time the network gives a bad answer, according to the rules:

$$\begin{aligned} (w_{t+1}, c_{t+1}) &= (w_t, c_t) && \text{if } \operatorname{sgn}(w_t \cdot x_t + c_t) = \epsilon_t \\ (w_{t+1}, c_{t+1}) &= (w_t + \epsilon_t x_t, c_t + \epsilon_t) && \text{else.} \end{aligned}$$

Novikoff has shown that if two classes of points are finite and separable, and if each example is given arbitrarily many times to the network, then the algorithm converges in finite time. He also gives an explicit upper bound on the maximal number of correcting steps. This bound is expressed in terms of the radius of the set of examples

$$R = \max\{|x| : x \in \mathcal{C}^+ \cup \mathcal{C}^-\}$$

and the separation margin

$$\rho = \frac{1}{\inf\{\|w\| : w \cdot x + 1 > 0 \forall x \in \mathcal{C}^+ \text{ and } w \cdot x - 1 < 0 \forall x \in \mathcal{C}^-\}}$$

Theorem 7.1 (NOVIKOFF, 1962)

Rosenblatt's algorithm initialized with $w_0 = 0$ and $c_0 = 0$ converges in finite time, after at most $\lfloor R^2/\rho^2 \rfloor$ corrections.

7.3 Wavelets as neural transfer functions for single layer networks

Pati and Krishnaprasad noticed that a single layer neural network with a sigmoidal transfer function can be equivalent to a wavelet network. They thus suggested to use a single layer neural network with discrete wavelet transfer functions, so that the network function is

$$f(\boldsymbol{x}) = \sum_k c_k \psi(A_k \boldsymbol{x} + b_k) \tag{7.2}$$

where the coefficients to tune are the c_k's.

They then suggested two network growth rules, based either on the spectral analysis of the example set or on interscale wavelet coefficient correlation. The network weight tuning

is done by inverting a linear regression system, since the output depends linearly on these weights.

Zhang and Benveniste [ZB92] have suggested an approach similar to that of Pati and Krishnaprasad, that consists in finding an approximation by adaptive subsampling of the continuous wavelet transform. Their network is also a single layer neural network, of function (7.2), but they allow the wavelet dilation and translation coefficients A_k and b_k to be tuned by backpropagation.

Later, Zhang [Zha97] suggested an adaptive approximation algorithm that is close to our approach, and which we detail in the next chapter. Benveniste *et al.* [DJB95] have developed Monte-Carlo approaches to function learning on the basis of scattered samples. This approach allows at the same time to prove nonlinear approximation results, and to build a learning algorithm that converges in probability.

7.4 Adaptive control

The neural network approach can be used for adaptive control of a dynamical system whose parameters are unknown, as has been developed and studied in detail by Slotine *et al.* [Slo86, SC86, SL90]. In the case when the dynamic system equation depends linearly on unknown parameters, we can find stable and convergent algorithms. "Unknown parameters" can either be a finite set of unknown constants as well as whole unknown function.

7.4.1 Finite dimensional adaptive control

Let us give consider a simple example: we want to move a known mass m bound to a spring of unknown stiffness k, and to which we can apply a chosen force u. The evolution equation of the system is then:

$$m\ddot{x} = -kx + u \tag{7.3}$$

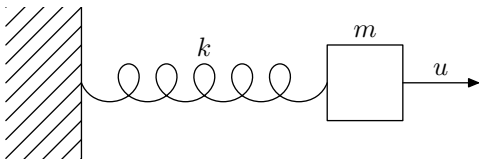


Figure 7.2: Description of the mass-spring system, with an input force u.

We now want the mass to follow a predefined trajectory $x_d(t)$. If the stiffness k is known, we simply choose the input u as:

$$u = kx + \left(\ddot{x}_d - 2\frac{\dot{x} - \dot{x}_d}{\tau} - \frac{x - x_d}{\tau^2}\right) \tag{7.4}$$

In this way, the evolution equation becomes

$$(\ddot{x} - \ddot{x}_d) + 2\frac{\dot{x} - \dot{x}_d}{\tau} + \frac{x - x_d}{\tau^2} = 0 \tag{7.5}$$

which guarantees that the tracking error $x - x_d$ can be written as some $(a + bt)e^{-t/\tau}$ and thus converges to 0 exponentially. It is usual to define the composite tracking error

$$s = \dot{x} - \dot{x}_d + \frac{x - x_d}{\tau} \tag{7.6}$$

This composite tracking error is chosen such that if it goes to 0, then the tracking error also goes to 0. It allows to consider the evolution equation as a differential equation of first order

$$\dot{s} = -\frac{s}{\tau} \tag{7.7}$$

If the spring stiffness is estimated to be $\hat{k}$, the tracking error does not go to 0 any more. The input becomes

$$u = \hat{k}x + \left(\ddot{x}_d - 2\frac{\dot{x} - \dot{x}_d}{\tau} - \frac{x - x_d}{\tau^2}\right) \tag{7.8}$$

and the evolution equation (7.7) written in the composite tracking error variable becomes

$$\dot{s} = -\frac{s}{\tau} + \frac{\hat{k} - k}{m}x \tag{7.9}$$

So far, the convergence of s to 0 is not guaranteed any more.

Adaptive control allows to choose a control law and an update rule for $\hat{k}$ that guarantee that the tracking error eventually goes to 0 and in some cases that $\hat{k}$ also converges to k.

We define for this the following functional

$$V = \frac{1}{2}s^2 + \frac{1}{2}\gamma(\hat{k} - k)^2$$

where γ is any chosen gain factor, and we assume that $\hat{k}$ is allowed to vary in time. This functional is a Lyapunov function that guarantees that the system converges if

- it is positive definite, i.e. it has a unique minimum for $s = 0$ and $\hat{k} = k$;
- it decreases in time.

The first condition is fulfilled by construction. We will see that we can choose an explicit evolution rule for $\hat{k}$ that guarantees that the Lyapunov function is nonincreasing in time. The time derivative of V is:

$$\begin{aligned}
\dot{V} &= s\dot{s} + \gamma(\hat{k} - k)\dot{\hat{k}} \\
&= -\frac{s^2}{\tau} + s\frac{\hat{k} - k}{m}x + \gamma(\hat{k} - k)\dot{\hat{k}} \\
\dot{V} &= -\frac{s^2}{\tau} + (\hat{k} - k)\left(\frac{s}{m}x + \gamma\dot{\hat{k}}\right)
\end{aligned} \tag{7.10}$$

The miracle is that by setting

$$\dot{\hat{k}} = -\frac{sx}{m\gamma} \tag{7.11}$$

we can cross out from expression (7.10) a term that was annoying for two reasons : it was able to make the Lyapunov function V increasing with time, and we could not estimate it. We thus obtain:

$$\dot{V} = -\frac{s^2}{\tau}$$

We can then show that with the control law (7.8) and with the update rule (7.11), the composite tracking error s goes to 0. Intuitively, it cannot stay strictly positive for a long time, as V would then become negative. Formally, this can be proved by using Barbalat's lemma.

We have no such strong result for $\hat{k}$. The existence of the Lyapunov function only proves that it is bounded. In practice, $\hat{k}$ converges to k only if it required for control purposes. If for example $x_d = 0$ and the system starts from the state $x(0) = 0$, $\dot{x}(0) = 0$, the input u will be zero, and $\hat{k}$ will stay constant, even if it is different from k. This is described as the "laziness" of the system, that only learns what is necessary. This is also the reason why it is not possible to give a quantitative convergence rate for s, as we can delay arbitrarily the estimation of k.

7.4.2 Function learning

This approach can be extended to any number of parameters, and even to an object which can be characterized with an infinite number of parameters: a function [SS92, SS92, CS95, SS95, CS95, BS97].

Let us assume that the system we want to control is of the form

$$x^{(n)} = f(x, \dots, x^{(n-1)}) + u$$

The measurable state of the system is composed of variables $x, \dots, x^{(n-1)}$. We define a target trajectory x_d we know with its derivatives of any order.

If we choose a discrete basis $(\phi_i)_{i\in\mathbb{I}}$ of $\mathbb{R}^n$, the unknown function f can be split into

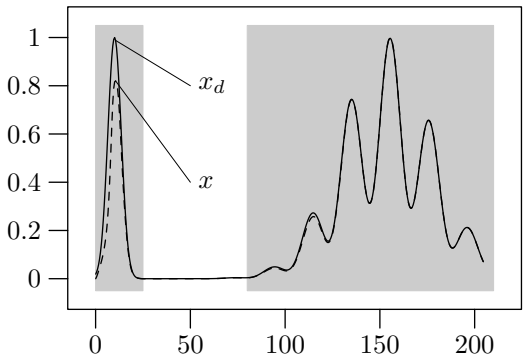
$$f(x, \dots, x^{(n-1)}) = \sum_{i\in\mathbb{I}} c_i \phi_i(x, \dots, x^{(n-1)})$$

and the learning problem is thus the problem of estimating the unknown coefficients c_i. If we choose a composite tracking error

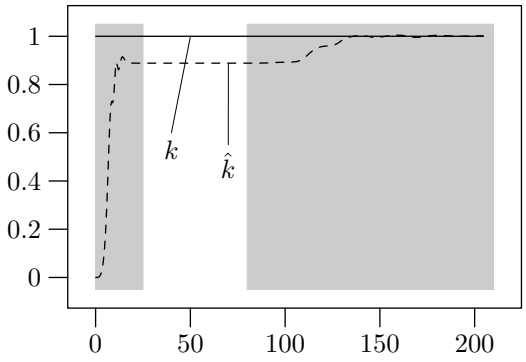
$$s = \left(\frac{d}{dt} + \frac{1}{\tau}\right)^{n-1} (x - x_d)$$

and a control law

$$u = -\sum_{i\in\mathbb{I}} \hat{c}_i \phi_i(x, \dots, x^{(n-1)}) + (\dot{x} - \dot{s}) - \frac{s}{\tau}$$



(a) Desired and real trajectory



(b) Real stiffness k and estimated stiffness $\hat{k}$

Figure 7.3: Example of adaptive control for the mass-spring system of figure 7.2. We can see that the shaded areas where motion is required are those when the error estimation $k - \hat{k}$ of k decreases. Otherwise, the estimation $\hat{k}$ does not change.

we obtain the evolution equation in the composite tracking error

$$\dot{s} = -\frac{s}{\tau} + \sum_{i \in \mathbb{I}} (c_i - \hat{c}_i)\phi_i \left(x, \ldots, x^{(n-1)}\right)$$

We should remark that in the expression of u, the term $\dot{x} - \dot{s}$ can be expressed with derivatives of the real trajectory only up to the order $n-1$ (by crossing out a difference of two equal terms) and can thus be computed from the state.

If we now choose a set of gains $(\gamma_i)_{i \in \mathbb{I}}$, the Lyapunov function

$$V = \frac{1}{2}s^2 + \frac{1}{2}\sum_{i \in \mathbb{I}} \gamma_i (c_i - \hat{c}_i)^2$$

has the time derivative

$$\dot{V} = -\frac{1}{\tau}s^2 + \sum_{i \in \mathcal{I}} (c_i - \hat{c}_i)(s\phi_i(x, \ldots, x^{(n-1)}) - \gamma_i \hat{c}_i)$$

The Lyuponov therefore decreases in time if we choose the update rule on the coefficients $\hat{c}_i$:

$$\dot{\hat{c}}_i = \frac{s\phi_i(x, \ldots, x^{(n-1)})}{\gamma_i}$$

Remarks

- *This learning scheme is one of the very few that fulfills the strong requirements of control stability.*
- *We can see that the coefficient estimate evolution is similar to a stochastic gradient or backpropagation update rule. Each coefficient is modified to the extent that it influences the output (which is $\phi_i(x, \ldots, x^{(n-1)})$) and proportionally to the error.*
- *We will see that the choice of the parameters γ_i is essential, especially when the number of parameters c_n may be infinite. The choice of these gains consists in choosing a smoothness measure on the learned function. If for example the expansion basis is an orthogonal wavelet basis (ψ_{jk}) and if the corresponding gains γ_{jk} are of the form*

  $$\gamma_{jk} = 1 + 2^{2\alpha j}$$

  the Lyapunov function contains a smoothness measure with is a Sobolev error norm of index α.

  Thus, the Lyapunov function fits again in the classical framework: a matching functional and a matching and smoothness functional. In this particular case, the second term has a hybrid meaning: it imposes a given smoothness to the function, and is also an matching functional. This approach shares some features with regularization which is described in the next section.

7.5 Regularization and radial basis functions

Regularization is an approximation and interpolation framework that is theoretically well founded. This technique has undergone several recent evolutions, under which we can cite Vapnik's support vector machines, and the fast evaluation techniques developed by Beatson and Newsam [BN98].

Regularization [TA77] consists in choosing a Hilbert space $\mathcal{H}$ of functions mapping $\mathbb{R}^d$ to $\mathbb{R}$, with a norm $\|.\|_{\mathcal{H}}$ and the related inner product $\langle .,. \rangle$. The interpolation constraints have to be equicontinuous w.r.t this norm:

$$\exists M,\, \forall x,\, \forall f \in \mathcal{H},\, |f(x)| \leq M \|f\|_{\mathcal{H}}$$

7.5.1 Regularized problem

The regularized interpolating solution is the function f of minimum norm that fulfills the interpolation constraints, i.e.

$$f = \arg\min\{\|f\|_{\mathcal{H}} : f(x_n) = y_n,\, \forall n = 1 \dots N\} \tag{I}$$

A more general form of a regularized formulation consists in minimizing a criterion that combines constraints of smoothness and of adequacy to the interpolation constraints, of the form

$$f = \arg\min_{f \in \mathcal{H}} \sum_{n=1}^{N} |f(x_n) - y_n|^2 + \lambda \|f\|_{\mathcal{H}}^2 \tag{II}$$

The second problem is always well posed, because it is a minimization problem of a quadratic positive definite functional in a Hilbert space that has thus a unique solution. The first problem is only well posed if the interpolation constraints are not degenerate. This requires that the measure points should be distinct and that the Hilbert space $\mathcal{H}$ should be "rich" enough. "Rich" means that for example for any finite subset $X \subset \mathbb{R}^d$, we should have

$$\{f|_X : f \in \mathcal{H}\} = \mathbb{R}^X$$

This assumption cannot be fulfilled by a finite dimensional set. It is however fulfilled by any Sobolev space of continuous functions.

We will see later in this chapter that the setting (II) of the problem consists in assuming that the measures y_i are contaminated with a additive Gaussian noise of variance proportional to λ. The formulation (I) is then the limit case corresponding to the assumption of noise of variance zero.

7.5.2 Solving the regularized problem

The following equations are not supposed to prove already well known results, and are not presented with detailed theoretic assumptions that a more formal description would require.

In a Hilbert space, a linear continuous functional can be written as the scalar product with a vector (Riesz's representation theorem). Thus, for all $x \in \mathbb{R}^d$, there exists a vector $v_x \in \mathcal{H}$ such that

$$f(x) = \langle f, v_x \rangle \qquad \forall f \in \mathcal{H}$$

We also define the interpolation kernel

$$K(x, y) = \langle v_x, v_y \rangle.$$

By construction, we have $K(x, y) = \overline{K(y, x)}$.

The solution of problem (II) is then written explicitly with the vectors v_x. Indeed, by setting

$$\begin{aligned} M[f, \lambda] &= \sum_{n=1}^{N} |f(x_n) - y_n|^2 + \lambda \|f\|_{\mathcal{H}}^2 \\ &= \sum_{n=1}^{N} |\langle f, v_{x_n} \rangle - y_n|^2 + \lambda \langle f, f \rangle \end{aligned}$$

the differential dM is

$$\begin{aligned} dM &= 2\lambda \langle f, df \rangle + 2 \sum_{n=1}^{N} \langle df, v_{x_n} \rangle \langle f, v_{x_n} \rangle - 2 \sum_{n=1}^{N} y_n \langle df, v_{x_n} \rangle \\ &= \left\langle 2\lambda f + 2 \sum_{n=1}^{N} \langle f | v_{x_n} \rangle v_{x_n} - 2 \sum_{n=1}^{N} y_n v_{x_n} \middle| df \right\rangle \end{aligned}$$

At the minimum, this differential is zero, which means

$$\lambda f = \sum_{n=1}^{N} (y_n - \langle f, v_{x_n} \rangle) v_{x_n} \tag{7.12}$$

We can already read from this formula that the solution is a linear combination of N functions v_{x_n}. We set

$$f = \sum_{n=1}^{N} c_n v_{x_n} \tag{7.13}$$

$$\text{i.e.} \quad f(x) = \sum_{n=1}^{N} c_n K(x_n, x)$$

and

$$C = \begin{bmatrix} c_1 \\ \vdots \\ c_N \end{bmatrix}, \qquad K = \begin{bmatrix} \langle v_{x_1}, v_{x_1} \rangle & \cdots & \langle v_{x_1}, v_{x_N} \rangle \\ \vdots & & \vdots \\ \langle v_{x_N}, v_{x_1} \rangle & \cdots & \langle v_{x_N}, v_{x_N} \rangle \end{bmatrix},$$
$$F = \begin{bmatrix} \langle f, v_{x_1} \rangle \\ \vdots \\ \langle f, v_{x_N} \rangle \end{bmatrix} \qquad \text{and } Y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}.$$

We obtain by inner product of (7.13) and (7.12) with the v_{x_m} for $m = 1 \dots N$ the relations

$$KC = F \tag{7.14}$$
$$\lambda F = K(Y - F) \tag{7.15}$$

from which we infer

$$C = (K + \lambda I)^{-1} Y \tag{7.16}$$

In the limit case where $\lambda \to 0$, we must also assume that K is invertible, which implies that the x_n should all be different, and we can write

$$C = K^{-1} Y \tag{7.17}$$

which is the solution of (I).

7.5.3 Reproducing regression kernels

The reproducing kernel [Aro50, GJP95] associated with this Hilbert space is the function

$$K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$$
$$(x, y) \mapsto v_x(y) = \langle v_x, v_y \rangle$$

We call this kernel "reproducing" because it fulfills the following relationships:

$$\langle h, K(., x) \rangle = h(x) \qquad \forall h \in \mathcal{H},\ x \in \mathbb{R}^d$$
$$\langle K(., x), K(., y) \rangle = K(x, y) \qquad \forall x, y \in \mathbb{R}^d$$

The kernels we have seen until now are positive definite, because the symmetric collocation matrix K, which can be written

$$K = \begin{bmatrix} K(x_1, x_1) & \dots & K(x_1, x_N) \\ \vdots & & \vdots \\ K(x_N, x_1) & \dots & K(x_N, x_N) \end{bmatrix}$$

is always positive definite if the points x_n are all different.

The most usual reproducing kernels are translation invariant reproducing kernels:

$$K(x+\delta, y+\delta) = K(x,y) \qquad \forall x, y, \delta$$

that can be written in a simplified form as $K(x,y) = K(x-y)$. The reproducing kernels for Sobolev spaces with a translation invariant norm over an infinite domain (like $\mathbb{R}^d$) are translation invariant. The Hilbert norm can the be written as

$$\|f\|_{\mathcal{H}} = \int_{\mathbb{R}^d} \frac{|\hat{f}(\omega)|^2}{\hat{K}(\omega)} d\omega$$

where the Fourier transform $\hat{K}$ of K is integrable, strictly positive and bounded:

$$\hat{K}(\omega) \in]0, M] \qquad \forall \omega$$
$$\int_{\mathbb{R}^d} \hat{K}(\omega) d\omega < +\infty$$

Among these kernels, it is also usual to use isotropic kernels, which are finally written

$$K(x-y) = g(\|x-y\|)$$

We can note that in this case, the kernel K and its Fourier transform are radial functions.

7.5.4 Positive definite functions

In the radial case, the interpolation kernel is then written

$$K(x,y) = g(\|x-y\|)$$

According to the above results, the function g is suitable if the Fourier transform of the function $x \mapsto g(\|x\|)$ mapping $\mathbb{R}^d$ to $\mathbb{R}$ is positive. The criterion involving explicitly the dimension d, the function g is not guaranteed to be positive definite in all dimensions.

A theorem of Schoenberg gives a characterization of functions g that can be used as a radial basis function profile for any space dimension d. We state it shortly by giving some definitions.

Definition 7.1

*A function g mapping $(0, +\infty)$ to $\mathbb{R}$ is said to be **positive definite** if for all integers d and N, for any N-uple $(x_1, \ldots, x_N)$ of different points of $\mathbb{R}^d$, the matrix of coefficients $g(\|x_n - x_m\|)$ is* positive definite.

Definition 7.2

*A function h from $(0, +\infty)$ to $\mathbb{R}$ is said **completely monotonic** if*

1. *it is infinitely differentiable*
2. *for all $t \in (0, +\infty)$ and for all integer n*

$$h^{(n)}(t) > 0 \qquad \text{if } n \text{ is even,}$$
$$h^{(n)}(t) < 0 \qquad \text{if } n \text{ is odd.}$$

We can now state

Theorem 7.2 (Schoenberg, 1950)

The function g is positive definite if and only if the function $t \mapsto g(\sqrt{t})$ is completely monotonic.

This theorem has two considerable advantages: we do not need to compute any high dimensional Fourier transform, and it gives a simple way to choose a function g which is a suitable radial basis function profile for any dimension.

We can thus give the following pairs of completely monotonic functions and of positive definite functions:

Completely monotonic	Positive definite
$x \mapsto e^{-x/2\sigma^2}$	$r \mapsto e^{-r^2/2\sigma^2}$
$x \mapsto \frac{1}{(a^2+x)^\alpha}$	$r \mapsto \frac{1}{(a^2+r^2)^\alpha}$

This model allows us to build functions that are positive definite at any dimension. However, we thus forbid ourselves to use functions that may work only at a given dimension.

7.5.5 Bayesian model

We can motivate such an approach with a Bayesian model, that relies on an a priori model of the signal and of the measurement noise. We first assume that the function f to estimate is a realization of a random process of density

$$dP(f) \propto e^{-\|f\|^2_{\mathcal{H}}/2s^2}$$

This apparently tailor-made assumption consists in fact in doing the following hypothesis on the function values:

1. $f(x)$ is a Gaussian centered variable,
2. $E(f(x)f(y)) = s^2K(x,y)$

and the kernel thus explicitly appears as a correlation term between pointwise values of the function[1].

We then assume that the y_n are noisy measures:

$$y_n = f(x_n) + \epsilon_n$$

where the variables ϵ_n are i.i.d Gaussian variables of standard deviation σ. We can then write the conditional probability density of y_n w.r.t the function f

$$dP(\{y_n\}|f) \propto \prod_{n=1}^{N} e^{-(f(x_n)-y_n)^2/2\sigma^2}$$

[1] Parzen has shown in 1962 that the choice of a reproducing kernel Hilbert space and that of a centered stochastic process of covariance $K(x,y)$ are equivalent (if K is symmetric positive definite).

Bayes' rule consists in looking for a function f that maximizes the a posteriori conditional probability density $dP(f)$ w.r.t the measures y_n:

$$\begin{aligned} dP(f|\{y_n\}) &\propto \frac{dP(\{y_n\}|f)dP(f)}{dP(\{y_n\})} \\ &= e^{-\|f\|^2_{\mathcal{H}}/2s^2} \times \prod_{n=1}^{N} e^{-\epsilon_n^2/2\sigma^2} \\ &\propto \exp\left(-\frac{\|f\|^2_{\mathcal{H}}}{s^2} - \sum_{n=1}^{N}(f(x_n)-y_n)^2/2\sigma^2\right) \end{aligned}$$

which consists in minimizing

$$\lambda\|f\|^2_{\mathcal{H}} + \sum_{n=1}^{N}(f(x_n)-y_n)^2$$

whenever λ is the variance ratio of noise and signal:

$$\lambda = \left(\frac{\sigma}{s}\right)^2$$

We see that the main difference between both problems (I) and (II) : the first form corresponds to the assumption that the measure noise is zero. We also can see that choosing a weighting coefficient λ consists in doing an a priori assumption on the relative importance of the noise and the signal.

7.5.6 Limits of the regularized approach

When we choose a regularization norm, we have to choose a function K such that the norm can be written as

$$\int \frac{|\hat{f}(\omega)|^2}{\hat{K}(\omega)}d\omega$$

where K has to decay as ω goes to $+\infty$ and take a bounded value in 0. The choice of such a function includes the choice of an a priori scale. We can extract this a priori scale from the function in the following way. We set:

$$\Delta\omega = \inf\{|\omega| : f(\omega) < f(0)/2\}$$

The scale parameter is then $\Delta x = 1/\Delta\omega$. This approach is thus not scale invariant, and the choice of the scale parameter has to be done on the basis of the measure point density. If this density is not uniform, we have either to find a compromise value of the scale parameter or to give up the beautiful theoretic framework of regularization.

7.6 Extension to semi-norms and to conditionally positive definite kernels

If we want to choose a quadratic functional that penalizes the singular behavior of a function without introducing a scale parameter, we have to choose a kernel K built with a homogeneous function g, like $r \mapsto r^\alpha$. We then obtain seminorms:

$$\|f\|_{\mathcal{H}}^2 = \int |\hat{f}(\omega)| \times \|\omega\|^\alpha d\omega$$

7.6.1 Representation theorem

If the function $\omega \mapsto \hat{K}(\omega)$ as $\omega \to +\infty$, the seminorm

$$\|f\|_{\mathcal{H}} = \int_{\mathbb{R}^d} \frac{|\hat{f}(\omega)|^2}{\hat{K}(\omega)} d\omega$$

does not define a Hilbert space any more, a semi-Hilbert space, which can be seen as the sum of

- the space of tempered distributions whose Fourier transform is a function which is locally square integrable over $\mathbb{R}^d - \{0\}$ and for which the above seminorm integral is finite,
- a space $\mathcal{N}$ of finite dimension of the tempered distributions f such that $\hat{f}(\omega) \times \sqrt{\hat{K}(\omega)}$ has is well defined and 0. In practice, the pole of $\hat{K}(\omega)$ is of polynomial divergence and this space is a set of polynomials.

The above presented results can be extended to this case. We can cite a theorem of Kimeldorf and Wahba which is:

Theorem 7.3 (Kimeldorf,Wahba)

The problem of minimizing the quadratic functional

$$f \mapsto \int_{\mathbb{R}^d} |\hat{f}(\omega)|^2 / K(\omega) d\omega$$

with the constraints

$$f(x_n) = y_n \qquad \text{pour } n = 1 \dots N$$

is well posed in a semi-Hilbert space provided that

$$(P \in \mathcal{N} \text{ and } \forall n,\, P(x_n) = 0) \Rightarrow P = 0$$

The solution can be then characterized the following way. It is written as

$$f(x) = P + \sum_{n=1}^{N} c_n K(x_n, x) \qquad \text{where } P \in \mathcal{N},$$

it fulfills the interpolation constraints

$$f(x_n) = y_n$$

and the coefficients (c_n) *fulfill*

$$\sum_{n=1}^{N} c_n P(x_n) = 0 \qquad \forall P \in \mathcal{N}$$

We can find a similar formulation for the noisy measure variant of the interpolation problem. We can show that such interpolation problems always have a unique solution if we choose a space of polynomials $\mathcal{N}$ and a symmetric kernel $x, y \mapsto K(x, y)$. The requirement that K should be positive definite is replaced with the requirement that it should conditionally positive definite:

Definition 7.3

A continuous symmetric function from $(\mathbb{R}^d)^2$ *to* $\mathbb{R}$ *is* ***conditionally positive definite*** *w.r.t a space of polynomials* $\mathcal{N}$ *if it fulfills the following condition: for all integer* N *and all points* $x_1, \dots x_N$ *of* $\mathbb{R}^d$*, and all reals* $c_1 \dots c_N$*, we have*

$$\sum_{n=1}^{N} \sum_{m=1}^{N} c_n c_m K(x_n, x_m) > 0$$

if

$$\sum_{n=1}^{N} c_n P(x_n) = 0 \qquad \forall P \in \mathcal{N}$$

and some c_n *are different from* 0*.*

The same way as for definite positive kernels, we can find functions g such that the radial kernel

$$\mathbb{R}^d \to \mathbb{R}$$
$$x \mapsto g(\|x\|)$$

is conditionally positive definite w.r.t to the space $K_{p-1}(\mathbb{R}^d)$ of polynomials of degree less than p, for any dimension d. Thus

Definition 7.4

A function from $[0, +\infty)$ *to* $\mathbb{R}$ *is conditionally positive definite of order* p *if for any integer* d*, for any integer* N*, any* N*-uple* $(x_1, \dots, x_N)$ *of points in* $\mathbb{R}^d$*, and any* N*-uple* $(c_1, \dots, c_N)$ *different from* $(0 \dots 0)$*, we have*

$$\sum_{n=1}^{N} \sum_{m=1}^{N} c_n c_m g(\|x_n - x_m\|) > 0$$

whenever the coefficients c_n fulfill

$$\sum_{n=1}^{N} c_n P(x_n) = 0 \qquad \forall P \in K_p(\mathbb{R}^d)$$

Schoenberg's theorem has been extended by Micchelli [Mic86] to conditionally positive definite functions.

Theorem 7.4 (Micchelli, 1986)

A function g is conditionally positive definite of order p if and only if the function

$$h : t \mapsto g(\sqrt{t})$$

fulfills

$$(-1)^k h^{(k)}(t) > 0 \qquad \forall k \geq p, \forall t > 0$$

We thus again have an exact characterization of functions g that generate conditionally positive definite radial kernels for spaces $\mathbb{R}^d$ of any dimension, and that is much simpler than Fourier considerations.

As an example, we can give the following functions

Functions h	Radial functions	Order p
$x \mapsto \sqrt{a^2 + x}$	$r \mapsto \sqrt{a^2 + r^2}$	1
$x \mapsto (a^2 + x)^\alpha$, $\alpha \notin \mathbb{N}$	$r \mapsto (a^2 + r^2)^\alpha$	$\lfloor \alpha \rfloor$
$x \mapsto x^{n+1/2}$	$r \mapsto r^{2n+1}$	n

Duchon has studied such a family of radial basis functions. He suggested the following regularization functional:

$$\|f\|_{\mathcal{H}} = \int |\hat{f}(\omega)|^2 \|\omega\|^{2m} d\omega$$

whose reproducing kernel has the Fourier transform

$$\hat{K}(\omega) = \frac{1}{\|\omega\|^{2m}}$$

The smoothness exponent m has to be large enough for the kernels to be continuous, which means that $m > d/2$.

The space expression of the kernel is then

$$K(x) = \|x\|^{2m-d} \qquad \text{if } d \text{ is odd}$$
$$K(x) = \|x\|^{2m-d} \ln \|x\| \qquad \text{else.}$$

The polynomial space associated to such functions is the space of polynomials of degree less than m.

7.6.2 Approximation order

A number of theorems allow to estimate the convergence rate of radial basis function network depending on the smoothness of the estimated function and of the smoothing norm we use, and are described for example in [SW98].

If we denote by f the unknown function, and by $f_{\mathcal{X}}$ the interpolant on the samples $(\boldsymbol{x}, f(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$, we can define:

The fill distance of $\mathcal{X}$ is the real

$$h(\mathcal{X}) = \max_{\boldsymbol{x}\in D} \min_{\boldsymbol{x}'\in\mathcal{X}} \|\boldsymbol{x} - \boldsymbol{x}'\|$$

The separation distance is the number

$$q(\mathcal{X}) = \min_{\substack{\boldsymbol{x}\in\mathcal{X},\boldsymbol{x}'\in\mathcal{X} \\ \boldsymbol{x}\neq\boldsymbol{x}'}} \|\boldsymbol{x} - \boldsymbol{x}'\|$$

If the function f is the restriction to an open bounded set $\Omega \subset \mathbb{R}^d$ of a function of the Sobolev space $H^{d/2+s}$, and if we take an interpolation kernel K_s such that

$$A\|\boldsymbol{\omega}\|^{-d-2s} \leq \hat{K}(\boldsymbol{\omega})| \leq B\|\boldsymbol{\omega}\|^{-d-2s} \tag{7.18}$$

then the approximation error $\|f - f_{\mathcal{X}}\|_\infty$ is bounded by

$$\|f - f_{\mathcal{X}}\|_\infty \leq Mh(\mathcal{X})^s$$

If the function f is not smooth enough, no convergence can be proved. Moreover, if the function smoothness varies locally, and the function is locally of Sobolev smoothness $H^{d/2+s'}$, we have to choose a kernel K_s of smoothness s low enough in order that $s < \min s'$.

This is double problem: we have to choose an a priori smoothness index s. If the estimated smoothness s' is too low, we obtain a low convergence rate, and if it is too high, we have no convergence. Moreover, this parameter is unique. If the function f has a non uniform smoothness, we have to tune the radial basis functions to the lowest smoothness $\min s'$, and thus have everywhere a low convergence rate of $h(\mathcal{X})^{\min s'}$.

This approach has another drawback: the bad conditioning of the system matrix of interpolation constraints. The norm of the inverse matrix is related to the separation distance of the set of measuring points (the smallest distance between two different measuring points). If we denote by A the system matrix, we have the only bound

$$\|A^{-1}\|_{\infty,\infty} \leq q(\mathcal{X})^{-s}$$

which diverges all the faster as the interpolation process is of high order, which is sometimes named the *uncertainty relation* of regularization.

7.6.3 Fast computations with radial basis functions

Normally, the computation at a single point of a linear combination of kernels:

$$f(x) = \sum_{n=1}^{N} c_n g(\|x_n - x\|)$$

requires N operations which the kernels are not localized, which is the case of almost all the kernels (expect for the Gaussian). As the number of kernels increases, the pointwise evaluations can get longer and longer.

Beatson and Newsam [BCN98, BN98] have suggested a fast evaluation scheme that consists in grouping in a hierarchical way kernel centers that are close one to another. Their approach is based on an algorithm developed by Greengard and Rokhlin. We shortly describe Greengard and Rokhlin's approach, and point out what Beatson and Newsam added to it.

Greengard and Rokhlin wanted to solve numerically an elliptic partial differential equation with a finite number of pointwise sources, and whose Green kernel is up to an elementary transformation an analytic function. They designed a fast method to evaluate potentials of the form

$$f(z) = \sum_{n=1}^{N} c_n \phi(z - z_n)$$

defined over the complex plane. The kernels ϕ are analytic kernels of the form

$$\phi : z \mapsto \frac{1}{z}$$
$$\text{or} \quad \phi : z \mapsto \ln z$$

Greengard and Rokhlin showed that if we group centers $(z_n)_{n \in P}$ that are close one to another, we can find a multipole expansion around a center z_c that approximates the contribution of these centers with a good accuracy at a distance large w.r.t the size of this group of centers:

$$\sum_{n \in P} c_n \phi(z - z_n) = a_{-1} \ln(z - z_c) + a_0 + \frac{a_1}{z - z_c} + \cdots + \frac{a_m}{(z - z_c)^m} + \mathcal{O}\left(\frac{r}{|z - z_c|}\right)^{m+1}$$

where the radius r is defined as

$$D = \max_{n \in P} |z_n - z_c|$$

They also develop calculus rules to shift multipole expansions, that is to expand a expression of the form

$$a_{-1} \ln(z - z_c) + a_0 + \frac{a_1}{z - z_c} + \cdots + \frac{a_m}{(z - z_c)^m}$$

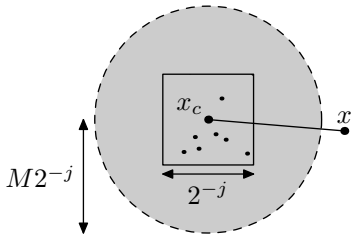


Figure 7.4: Domain of good accuracy of the multipole expansion approximation. If the centers $(z_n)_{n\in P}$ are located in a square of center z_c and of width 2^{-j}, the multipole approximation of the contribution of these centers is of good accuracy outside a circle of radius $M2^{-j}$.

into

$$b_{-1}\ln(z-z'_c)+b_0+\frac{b_1}{z-z'_c}+\cdots+\frac{b_m}{(z-z'_c)^m}$$

Their evaluation method can then be split into two steps. A preparatory step consists in placing the kernel centers z_n into a tree. The tree arity is 4, and each node is associated to a dyadic square of the plane. The tree structure is associated to the square subdivision into 4, i.e. the node associated to the square

$$2^{-j}[k;k+1)\times[k';k'+1)$$

has as sons the nodes associated to the squares

$$2^{-j-1}[2k+\epsilon;2k+1+\epsilon)\times[2k'+\epsilon';2k+1+\epsilon') \qquad (\epsilon,\epsilon')\in\{0,1\}^2$$

The tree is built in such a way that the leaves (the nodes associated to the smallest squares) contain a bounded number ν of centers z_n,

For each leaf, we compute the multipole expansion around the center of the corresponding square. Then, we compute the multipole expansion of intermediate nodes by tree ascent: the multipole expansions of the 4 son nodes are shifted and summed to make up the multipole expansion of the father node. At the end of this preparatory step, we have a tree of embedded squares for each of which we have a multipole expansion of the contribution of the centers in the square that is valid at a distance large w.r.t the square width, as shown in Fig. 7.4.

The computation of the potential at a given point then uses recursively this set of multiscale approximations. The potential value is computed as the sum of contributions of the largest scale squares. If the evaluation point is far away from a square, we use the square's multipole expansion to estimate the contribution of its centers. Else, we add separately the contribution of the four sons of the square, which is done recursively.

As the kernels used by Greengard and Rokhlin are analytic, the multipole expansions are formally the same as one dimension Laurent expansions.

Beatson and Newsam extend this approach to non analytic kernels. Asymptotic expansions in its more general setting includes differentials of any order k, i.e. tensors of 2^k

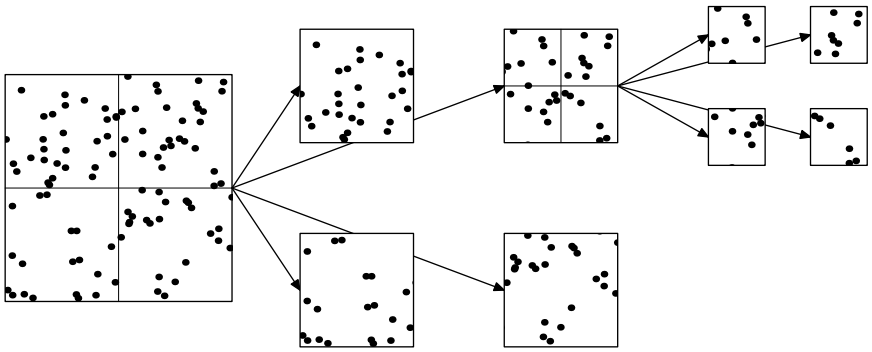


Figure 7.5: The Greengard tree for the fast evaluation of a potential. The figure shows the hierarchic structure connected to the square subdivision.

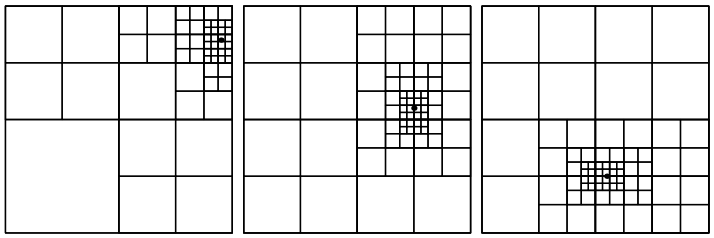


Figure 7.6: Example grids used to compute a potential at different points. We see that the squares we use are all the smaller as they are close to the evaluation point.

coefficients in 2 dimensions. Since however the kernels used are radial, this expansion can be simplified into an expansion of the form

$$\sum a_{p,q} \frac{(x.u)^p}{\|x\|^q} \qquad \text{where the } p\text{'s and } q\text{'s always fulfill } p < q$$

which are noticeably less easy to use that Greengard and Rokhlin's expansions. The validity is however far more general, as the kernels do not need to be analytic, and as the input space can be of any dimension.

For homogeneous kernels (that are conditionally positive definite), the approximation rate is of order $(r/\|x - x_c\|)^p$ where p is the highest degree in the expansion, which means that the distance of the evaluation point x to the center x_c of a square has to be large to the square width r.

$$\|x - x_c\| \gg r$$

Beatson and Newsam also apply this technique to non homogeneous kernels. As such kernels, we can for example mention the multiquadrics

$$x \mapsto \sqrt{a^2 + \|x\|^2}$$

where a parameter a (a distance) appears. In this case, the multipole expansion is valid only if

$$\|x - x_c\| \gg \sqrt{r^2 + a^2}$$

which imposes a limit on the resolution of the square subdivision, as it does not bring much to use squares of width much smaller than a.

Numerical results between a direct evaluation and a evaluation with multipole expansions reported by Beatson and Newsam are shown on table 7.1. They give the computation time to evaluate on 3200 different locations a sum of 3200 kernels placed at random locations. The approximation order is chosen in order to have a relative error of 10^{-10}. The theoretic computation speed of N values of an expansion of N kernels is $\mathcal{O}(N \log N)$, while a direct evaluation costs N^2 operations. Also note that the preparatory step is the most expensive, as the evaluation of N function values only costs $\mathcal{O}(N)$ operations.

Kernel type	$\sqrt{x^2 + .001}$	$x^2 \ln \lvert x \rvert$
Time (direct evaluation)	32.2 s	63.6 s
Time (multipole evaluation)	0.71 s	0.97 s

Table 7.1: Comparison between direct evaluation and multipole evaluation

This method thus allows to perform an expansion evaluation in $N \log N$ operations. It can also be used to solve the collocation system with an iterative procedure. Its complexity does not allow to use it with more complex or direct matrix inversion methods.

7.7 Vapnik's support vector machines

The support vector machines can be used in regularization approaches to solve nonparametric regression problems. Their interest is to provide a sparse solution, as in general most of the coefficients of the kernel expansion they provide are 0. The support vector approach is based on a more general theory on complexity and generalization ability we remind here.

Vapnik and Chervonenkis have described a complexity measure of set of functions that is called the Vapnik-Chervonenkis (VC) dimensions. Vapnik then proved that it is possible to bound with a probability arbitrarily close to 1 the approximation risk of the correspondence between two random variables X and Y. The bound is the sum of two known terms. The first term is the empirical risk, and the second term depends on a complexity measure of the family of functions from which we choose our approximation (its VC dimensions) and the number of samples N. We define the empirical risk as

$$R_{\text{emp}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(x_i))^2$$

and the real risk

$$R = E(Y - f(X))$$

The bound has the following form

$$R \leq R_{\text{emp}} + H(\eta, h, N) \qquad \text{with probability } 1 - \eta$$

where H is a known function.

On the basis of this result, Vapnik suggests to minimize the right hand side of the above inequality, which he calls the *structural minimization risk*

$$R_{\text{struct}} = R_{\text{emp}} + H(\eta, h, N)$$

A possible implementation of this principle consists in choosing an increasing sequence of function families $(S_n)_{n\in\mathbb{N}}$:

$$S_0 \subset S_1 \subset \cdots \subset S_n \subset S_{n+1} \subset \ldots$$

Their VC dimensions $\dim_{\text{VC}}(S_n)$ are increasing and we assume they are known. The shape of the curves R_{emp} and $H(\eta, \dim_{\text{VC}}(S_n), N)$ for a given η are respectively increasing and decreasing, and positive. The term $H(\eta, \dim_{\text{VC}}(S_n), N)$ diverges as the VC dimension goes to infinity, so there exists an optimal choice of an S_n for which the structural risk is minimum, as shown in Fig. 7.7.

7.7.1 Vapnik-Chervonenkis dimension

The Vapnik-Chervonenkis dimension (in short VC dimension) is a complexity measure we apply to a family of functions. We give a definition for it and compare it to the heuristic notion of generalization ability.

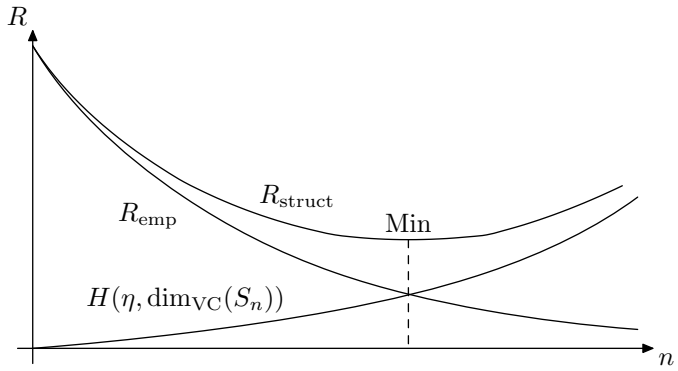


Figure 7.7: Principle of structural risk minimization

Definition 7.5 (VC dimension)

Let $f \in \mathcal{F}$ be a family of functions mapping a space X to the set $\{0,1\}$. The VC dimension of $\mathcal{F}$, denoted by $dim_{VC}\mathcal{F}$ is the largest integer N such that their exists N points $(x_1, \ldots, x_N) \in X^N$ for which

$$\{f|_{\{x_1 \ldots x_N\}} : f \in \mathcal{F}\} = \{0,1\}^{\{x_1 \ldots x_N\}}$$

This means that any binary function defined on $\{x_1, \ldots, x_N\}$ can be obtained as a restriction of an element of $\mathcal{F}$ to this finite subset. If this is possible for arbitrarily large integers N, the VC dimension for $\mathcal{F}$ is infinite.

This definition can be extended to real valued functions.

Definition 7.6

The VC dimension of a family $\mathcal{F}$ of functions mapping X to $\mathbb{R}$ is the dimension of the following family of binary functions:

$$\mathcal{F}' = \left\{x \mapsto 1_{(f(x)>\alpha)} : f \in \mathcal{F},\, \alpha \in \mathbb{R}\right\}$$

i.e. the dimension of the family of level functions we can build from these real valued functions.

Remarks

- *The VC dimension of a function family is at least N if there exists a N-uple of points such that the restriction of the family $\mathcal{F}$ to this N-uple makes up all possible binary functions. This is thus a measure of the highest possible complexity of the function family or of its worst case generalization capacity.*
- *It also worth noticing that this definition does not involve any structure (either topological or of other kind) of the space X. To illustrate this, let X be $[0,1]$ and let us*

choose a permutation on $[0,1]$ *as discontinuous as possible. If the family of functions* $f \in \mathcal{F}$ *has a given VC dimension, then the family of very irregular functions* $(f \circ \sigma)_{f \in \mathcal{F}}$ *has the same dimension.*

- *When* $\mathcal{F}$ *is a function family of* n *parameters (like on an* n *dimensional vector space), the VC dimension of this family is not always related to the number of parameters. It can be much smaller as well as much larger. Vapnik gives the pathologic example of the family*

$$(x \mapsto \sin(\alpha x))_{\alpha \in \mathbb{R}}$$

whose VC dimension is infinite. He shows that if we choose the points $x_1, \ldots, x_N$ *defined as*

$$x_1 = 10^{-1}, \quad \ldots, \quad x_N = 10^{-N}$$

any binary function b *defined over* $\{1, \ldots, N\}$ *can be represented as the restriction of* $x \mapsto \operatorname{sign} \sin(\alpha x)$ *to* $\{x_1, \ldots, x_N\}$ *if* α *is taken as*

$$\alpha = \pi \sum_{n=1}^{N} (1 - b(i)) 10^i + \pi$$

and this family of sine functions has thus a very bad generalization ability.

- *We can show that the VC dimension of a vector space of functions is equal to the vector space dimension of the space spanned by* F *and the constant function, thus either* $\dim F$ *or* $\dim F + 1$.

- *The VC dimension of a union of families* $\bigcup_k \mathcal{F}_k$ *is larger or equal that the largest VC dimension* $dim_{VC} \mathcal{F}_k$ *(in a similar way to the Hausdorff dimension, for example). However, it may happen that the VC dimension of a finite union* $\bigcup_k \mathcal{F}_k$ *is strictly larger than the largest VC dimension of the families* $\mathcal{F}_k$ *(which is not the case for the Hausdorff dimension).*

7.7.2 Bound on the approximation risk

The central theorem that provides a bound on the real risk as the sum of the empirical risk and of a complexity term has several variants depending on the problem to solve (classification or regression). In all cases, the key parameter is the VC dimension of the families of cost functionals associated to the chosen family of approximating functions.

If we approximate an unknown function $x \mapsto y = f(x)$ with an element of a family $(f_\alpha)_{\alpha \in A}$, the associated cost functional is

$$Q_\alpha : (x, y) \mapsto |y - f_\alpha(x)|^2$$

so that the real risk can be written

$$R(\alpha) = \int Q_\alpha(x, y) dP(x, y)$$

and the empirical risk

$$R_{\text{emp}}(\alpha) = \frac{1}{N}\sum_{n=1}^{N} Q_\alpha(x_n, y_n)$$

In theory, we do not have to control the complexity of the (f_α), but that of the associated cost functionals Q_α. In practice however, they are closely related, as

$$\dim_{\text{VC}}(f_\alpha)_{\alpha\in A} \leq \dim_{\text{VC}}(Q_\alpha)_{\alpha\in A} \leq 2\dim_{\text{VC}}(f_\alpha)_{\alpha\in A}$$

Theorem 7.5 (Vapnik, 1995)

Assume that the family of cost functionals Q_α are bounded by some A, then with probability larger than $1-\eta$, we have:

$$R(\alpha) \leq R_{emp}(\alpha) + A\sqrt{\frac{h(\ln\frac{2N}{h}+1) - \ln(\eta/4)}{N}} \qquad \forall\alpha \qquad (7.19)$$

where h is the VC dimension of Q_α, i.e.

$$h = dim_{VC}(Q_\alpha)_{\alpha\in A}$$

The information this theorem provides is twofold. We obtain qualitative information, that by increasing the model complexity we use to explain measures, we may have a lower empirical risk, but we definitely have a higher generalization risk, because

$$\sqrt{\frac{h(\ln\frac{2N}{h}+1) - \ln(\eta/4)}{N}}$$

increases with the VC dimension of the model. However, it also provides quantitative information because the terms of empirical risk and generalization risk are commensurable. This differs from approaches of classical regularization where the tradeoff between matching and capacity control is arbitrary. In the case of regularization, where the target functional is

$$R_{\text{emp}} + \lambda\|f\|^2_{\mathcal{H}}$$

arbitrary choices are made in the choice of the parameter λ and in the choice of the Hilbertian metric of the space $\mathcal{H}$.

Variants of this theorem exist for the cases when the cost functionals are not bounded (which is the case of regression, for example).

We take advantage of this relation by minimizing the right hand side of (7.19), which is called the structural risk, and which is the only term we are able to estimate:

$$R_{\text{struct}}(\alpha) = R_{\text{emp}}(\alpha) + A\sqrt{\frac{h(\ln\frac{2N}{h}+1) - \ln(\eta/4)}{N}}$$

7.7.3 Implementation of the structural risk minimization principle: support vector machines

As a general framework to minimize structural risk, Vapnik suggests the following approach. We suppose we have a increasing sequence of sets of functions $(S_n)_{n\in\mathbb{N}}$ of increasing VC dimensions. In a given S_n, one term of the structural risk is fixed: the complexity term. Minimizing the structural risk thus consists in minimizing the empirical risk. We thus obtain a solution $f_n \in S_n$. As n increases, the empirical risk decreases but the complexity term goes to $+\infty$. We can thus find an optimal n for which f_n minimizes the structural risk.

The choice of a given sequence S_n is the point where we introduce a priori information into the learning model.

The construction of support vector machines relies on the consideration of a particular type of classification : optimal hyperplane separation, which is the problem Rosenblatt's perceptron has to solve.

Vapnik shows that if two classes of points are separable with a hyperplane, separating them with a maximum margin hyperplane consists in minimizing the structural risk, because one can prove that the VC dimension of separating hyperplanes of margin ρ for a set of points of radius R is bounded by $\lfloor R^2/\rho^2 \rfloor$. For measure points $x_1, \ldots, x_n$ that of respective classes $\epsilon_1, \ldots, \epsilon_n$ where $\epsilon_i \in \{-1, +1\}$, the hyperplane of equation $y = w \cdot x + b$ is a separating hyperplane if

$$\begin{aligned} w \cdot x_i &> 0 && \text{if } \epsilon_i = +1 \\ w \cdot x_i &< 0 && \text{if } \epsilon_i = -1 \end{aligned}$$

and its margin is maximum if

$$\min_i \frac{|w \cdot x_i + b|}{\|w\|} \text{ is maximum}$$

This a quadratic programming problem, and its solution is written

$$w \cdot x + b = \sum_{i \in I} \alpha_i \langle x_i, x \rangle + b$$

where in practice the set I is relatively small, and the vectors $(x_i)_{i\in I}$ are called *support vectors*. Intuitively, the support vectors are the critical measurement points that are at the boundary of the classes to separate.

Hyperplane separation is not very useful since in most of the cases, the classes to separate are not linearly separable. To further take advantage of the properties of hyperplane separation, Vapnik suggests to make a linear separation, but not that of the points themselves. He suggests to separate linearly images of these points $X_i = \Psi(x_i)$ in a feature space of much higher dimension. The feature mapping Ψ is a nonlinear mapping chosen a priori.

It is then still possible to do an optimal separation of the features $X_i = \Psi(x_i)$. Since the resulting function can be exclusively written with inner products of the form $\langle X, X_i \rangle$, i.e. $\langle \Psi(x), \Psi(x_i) \rangle$, we never use the explicit definition of Ψ, but only the kernel K defined as

$$K(x, x') = \langle \Psi(x), \Psi(x') \rangle \tag{7.20}$$

If conversely we choose a symmetric positive definite kernel, Mercer's theorem states that there exists a feature space and a mapping Ψ, such that (7.20) is fulfilled.

To summarize, looking for a function expressed as

$$\epsilon(x) = \sum_{i \in I} c_i K(x_i, x) + c$$

such that the separation margin

$$\min_j \frac{|\sum_{i \in I} c_i K(x_i, x_j) + c|}{\sqrt{\sum_i c_i^2}}$$

be maximal consists in minimizing the structural risk. In the nonlinear case however, the problem is not a problem of quadratic programming any more. Fortunately, the dual optimization problem still is quadratic programming problem.

Support vector machines can also be used for regression. Let $|a|_\epsilon = \max(0, |a| - \epsilon)$. For measure points x_i, y_i, this consists in finding the function f of the form

$$f(x) = \sum_{i \in I} c_i K(x_i, x) \qquad \text{where } \|c\| \leq C$$

such that the functional

$$H[f] = \frac{1}{n} \sum_{i=1}^{n} |f(x_i) - y_i|_\epsilon$$

is minimal. The bound C and ϵ allow to control the generalization error. Some authors rewrite this problem as the global minimization of the functional

$$H[f] = \frac{1}{n} \sum_{i=1}^{n} |f(x_i) - y_i|_\epsilon + \sum_i c_i^2$$

In this case again, for a small subset $I \subset \{1 \dots n\}$ only, the coefficients $(c_i)_{i \in I}$ are nonzero, and the points $(x_i)_{i \in I}$ are called *support vectors*.

The technique of support vector machines is backed by a number of theoretical results, and provides stable estimations of limited complexity. However, it is very close to classical regularization, and thus inherits advantages and drawbacks from it. The main difference is that the resulting expansion is sparser than that provided by classical regularization.

However, the underlying signal representation is still the same. The function is sought as a linear combination of kernels centered around the measuring points. We have already seen that such representations are based on a strong a priori on the function smoothness and that no local adaptivity (in smoothness of in scale) is possible, because the kernels of the form $K(x_i, x)$ are all the same, up to a translation.

This leads us to base our approach on the choice of a representation, and only as a second step, to design an estimation algorithm.

Bibliography

[Aro50] A. Aronsajn. Reproducing kernel hilbert spaces. *Am. Math. Soc. Trans.*, 68:337–404, 1950.

[Bar93] A. R. Barron. Universal approximation bounds for superposition of a sigmoidal function. *IEEE Trans. on Information Theory*, 39(3):930–945, 1993.

[BCN98] R. Beatson, J. Cherrie and G. Newsam. Radial basis function methods for large data sets. In *International Workshop on CAGD and Wavelet Methods for Reconstructing Functions*. Montecatini, Italy, June 1998.

[BN98] R. K. Beatson and G. N. Newsam. Fast evaluation of radial basis functions: moment based methods. *SIAM J. Sci. Comput.*, 19(5):1428–1449, 1998.

[BS97] C. P. Bernard and J.-J. E. Slotine. Adaptive control with multiresolution bases. In *IEEE Conf. on Decision and Control (CDC'97), San Diego*. December 1997.

[CS95] M. Cannon and J.-J. E. Slotine. Space-frequency localized basis function networks for nonlinear system estimation and control. *Neurocomputing*, 9(3), 1995.

[Cyb89] G. Cybenko. Approximations by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 309(2), 1989.

[DJB95] D. Delyon, A. Judistky and A. Benveniste. Accuracy analysis for wavelet approximations. *IEEE Trans. Neural Networks*, 6(2):332–348, 1995.

[GJP95] F. Girosi, M. Jones and T. Poggio. Regularization theory and neural networks architectures. *Neural Computation*, 7(2):219–269, 1995.

[JZD+94] A. Juditsky, Q. Zhang, B. Delyon, P.-Y. Glorennec and A. Benveniste. Wavelets in identification wavelets, splines, neurons, fuzzies : how good for identification. Technical Report RR-2315, INRIA, France, 1994.

[Mic86] C. A. Micchelli. Interpolation of scattered data: Distance matrices and conditionally positive definite functions. *Constructive Approximation*, 2:11–22, 1986.

[SC86] J.-J. E. Slotine and J. A. Coetsee. Adaptive sliding controller synthesis for nonlinear systems. *International Journal of Control*, 3:44–47, 1986.

[SL90] J.-J. E. Slotine and W. Li. *Applied Non-Linear Control*. Prentice-Hall, 1990.

[Slo86] J.-J. E. Slotine. On modeling and adaptation in robot control. In *IEEE International Conference on Robotics and Automation, San Francisco, April 7-10, 1986*, pages 1387–1392. 1986.

[SS92] R. M. Sanner and J.-J. E. Slotine. Gaussian networks for direct adaptive control. *IEEE Transactions on Neural Networks*, 3(6):837–863, November 1992.

[SS95] R. M. Sanner and J.-J. E. Slotine. Structurally dynamic wavelet networks for the adaptive control of uncertain robotic systems. In *Proc. IEEE Conf. Decision and Control, New Orleans*. 1995.

[SSB+96] B. Schoelkopf, K. Sung, C. Burges, F. Girosi, P. Niyogi, T. Poggio and V. Vapnik. Comparing support vector machines with gaussian kernels to radial basis function classifiers. Technical Memo AIM-1599, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, December 1996.

[SW98] R. Schaback and H. Wendland. Inverse and saturation theorems for radial basis function interpolation. Preprint NAM 05, Universität Göttingen, 1998.

[SZL+95] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P.-Y. Glorennec, H. Hjalmarsson and A. Juditsky. Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31:1691–1724, December 1995.

[TA77] A. N. Tikhonov and V. I. Arsenin. *Solutions of Ill–Posed Problems*. W.H. Winston & Sons, John Wiley & Sons, Washington DC, 1977.

[Vap95] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer Verlag, New York, 1995.

[Vap97] V. N. Vapnik. The support vector method. *Lecture Notes in Computer Science*, 1327, 1997.

[ZB92] Q. Zhang and A. Benveniste. Wavelet networks. *IEEE Trans. on Neural Networks*, 3(6):889–898, November 1992.

[Zha97] Q. Zhang. Using wavelet network in nonparametric estimation. *IEEE Trans. Neural Networks*, 8(2):227–236, March 1997.

Chapter 8

Multiresolution interpolation

Abstract

In this chapter, we describe a new way to build a wavelet interpolant from scattered samples. It consists in selecting a subset of the wavelet basis and to solve a linear system of constraints. The central part is the selection step, because it determines the stability of the estimation. We therefore suggest an interpolation process that can be split in three steps.

1. The choice of a subfamily of a predefined wavelet basis, on the basis of the measurement locations. This is done with an *allocation algorithm* that selects these wavelet by building a one-to-one mapping between measures and wavelets. This steps selects as many wavelets as the measure points we are provided.
2. A second step consists in selecting from these wavelet/measure pairs a subset for which the resulting linear system of interpolation constraints is stable. We design for this a simple *geometric criterion* we call *exclusion criterion* in 1D and *good relative positioning* (GRP) criterion in the general case. This step thus discards some wavelet/measure pairs.
3. The last step is a simple inversion of the resulting square (and sparse) linear system.

We are then able to prove stability and convergence results that compare very favorably to regularization. The approximation order is locally adaptive, and the collocation matrix is very well conditioned. The first section of this chapter is devoted to the motivation of the use of a wavelet basis subfamily as a way to represent estimated functions.

8.1 Choice of a representation

To store or access an estimated function, we have to choose a way of representing it. Neural networks, or multilayer perceptrons represent a function as a combination of two building blocks: sigmoidal transfer functions and linear combinations of neuron outputs. Since the represented function does not depend linearly on the parameters like the synaptic coefficients,

little can be said about the convergence of learning algorithms. Moreover, nothing much more sophisticated than a gradient method can be used as a learning algorithm.

Our way of representing an estimated function will be definitely simpler : a finite linear combination of functions chosen from a given basis. Given N measures $(x_n, y_n)_{n=1...N}$, we seek an interpolant f as a linear combination of a finite subfamily $(g_n)_{n\in I}$ of a functional basis $\mathcal{G} = (g_n)_{n\in\mathbb{N}}$:

$$f = \sum_{n\in I} c_n g_n$$

Once we have chosen the subfamily indices I, the expansion coefficients are chosen by solving the interpolation constraints. The main subject of this discussion is therefore to decide how to sensibly choose the family I. The methods we have reviewed in the preceding chapter as well as Vapnik's structural risk minimization principle suggest to choose the most compact possible representation, which consists in taking an index set I of minimum size.

The choice of a basis $\mathcal{G} = (g_n)_{n\in\mathbb{N}}$ depends on two criteria:

- the basis $\mathcal{G}$ must have good approximation properties for the class of functions the unknown function f is supposed to belong to, which means that small subexpansions of functions f over $\mathcal{G}$ must be close to f;
- the computation time to evaluate a function value at a given point as well as the learning algorithm complexity must be small. This suggests to use localized functions, and *not* functions like complex exponentials $x \mapsto e^{ik\cdot x}$.

We will therefore start by an overview of some results in approximation theory to guide this choice.

8.1.1 Linear approximation

Linear approximation consists in replacing a target function f by an approximation f_N whose representation is simpler, with an approximation error $\|f - f_N\|$ as low as possible for a given coding size N.

If f is a periodic function, a $2N+1$ term approximation of f is for example its truncated Fourier series.

$$f_{2N+1} = \sum_{n=-N}^{+N} \langle f, e_n\rangle e_n$$
$$\text{where } e_n(x) = e^{inx}$$

We know that if f is uniformly α–Lipschitz, its Fourier coefficients fulfill the following inequality

$$\langle f, e_n\rangle = o(n^{-\alpha})$$

and the approximation error is thus

$$\|f - f_{2N+1}\|_2 = o(n^{1/2-\alpha})$$

We denote the N–term approximation error of a given function f in an orthonormal basis $\mathcal{G} = (g_n)_{n\in\mathbb{N}}$:

$$\epsilon_N^{lin}[f] = \left\| f - \sum_{n=1}^{N} \langle f, g_n \rangle g_n \right\|$$

This suggests the following definition we pick up from [DeV98a]

Definition 8.1

Let $\mathcal{G} = (g_n)_{n\in\mathbb{N}}$ be a Hilbert basis of a Hilbert space $\mathcal{H}$; the approximation space of index α denoted $\mathcal{A}_{lin}^{\alpha}$ is the set of all functions $f \in \mathcal{H}$ such that the approximation error decays in $N^{-\alpha}$:

$$f \in \mathcal{A}_{lin}^{\alpha} \Leftrightarrow \exists B, \left\| f - \sum_{n=1}^{N} \langle f, g_n \rangle g_n \right\| \leq B \times N^{-\alpha}$$

We can take as the norm of f in this new space the smallest B such that the above inequality is fulfilled, i.e. we set

$$\|f\|_{\mathcal{A}_{lin}^{\alpha}} = \max_{N\in\mathbb{N}} \left(\epsilon_N[f] N^{\alpha}\right)$$

We can reformulate the above remark by saying that the 2π–periodic α–Lipschitz functions are in the approximation space $\mathcal{A}_{lin}^{\alpha-1/2}$ if the Hilbert basis $\mathcal{G}$ is the Fourier basis.

We can prove theorems that state that approximation classes in the Fourier basis are functions of uniform regularity. Namely, we can embed an approximation space between two Sobolev spaces:

$$W^{\alpha} \subset \mathcal{A}_{lin}^{\alpha} \subset W^{\alpha-1/2-\varepsilon} \qquad \text{for all } \varepsilon > 0$$

A function is therefore in the approximation space $\mathcal{A}_{lin}^{\alpha}$ if the decay of its Fourier coefficients is fast enough.

8.1.2 Nonlinear approximation and Besov spaces

Doing an N–term nonlinear approximation of a given function f in a given basis $\mathcal{G} = (g_n)_{n\in\mathbb{N}}$ does not consist in truncating its expansion to the first N terms of its expansion in $\mathcal{G}$, but in keeping the N coefficients such that the approximation error be as small as possible, i.e. we set:

$$f_N^{nl} = \sum_{n\in I} \langle f, g_n \rangle g_n$$

where I is chosen such that $\operatorname{card} I = N$ and

$$\operatorname{card} I' = N \Rightarrow \left\| f - \sum_{n \in I} \langle f, g_n \rangle g_n \right\| \leq \left\| f - \sum_{n \in I'} \langle f, g_n \rangle g_n \right\|$$

When $\mathcal{G}$ is an orthonormal basis, this consists in taking the subexpansion with the N largest coefficients in the expansion of f.

If we rearrange the basis vectors g_n into g_{n_k} such that the sequence $|\langle f, g_{n_k} \rangle|$ is nonincreasing, the N term nonlinear approximation of f is

$$f_N = \sum_{k=1}^{N} \langle f, g_{n_k} \rangle g_{n_k}$$

We also define the nonlinear approximation error in the basis $\mathcal{G}$

$$\epsilon_N[f] = \left\| f - \sum_{k=1}^{N} \langle f, g_{n_k} \rangle g_{n_k} \right\|$$

This suggests the following definition for a nonlinear approximation space:

Definition 8.2

Let $\mathcal{H}$ be a Hilbert space and $\mathcal{G} = (g_n)_{n \in \mathbb{N}}$ an orthonormal basis of this space; the nonlinear approximation space of index α, written $\mathcal{A}^\alpha$ is defined by

$$f \in \mathcal{A}^\alpha \Leftrightarrow \exists B,\ \epsilon_N[f] \leq B \times N^{-\alpha}$$

$\mathcal{A}^\alpha$ is a vector space.

A more detailed definition of nonlinear approximation spaces can be given with three indices α, p and q.

Definition 8.3

Let $\mathcal{H}$ be a Hilbert space and $\mathcal{G} = (g_n)_{n \in \mathbb{N}}$ an orthonormal basis. We define the nonlinear N–term L_p approximation error as

$$\epsilon_{N,p}[f] = \inf_{\substack{\#I = N \\ (c_n)_{n \in I}}} \left\| f - \sum_{n \in I} c_n g_n \right\|_p$$

The nonlinear approximation space of indices α, p and q denoted by $\mathcal{A}_q^\alpha(L_p)$ is the set of functions f such that

$$\sum_{N=1}^{+\infty} \frac{1}{N} \left(N^\alpha \epsilon_{N,p}[f]\right)^q < +\infty \qquad \text{if } q < +\infty$$

$$\sup_{N \in \mathbb{N}^*} \left(N^\alpha \epsilon_{N,p}[f]\right) < +\infty \qquad \text{if } q = +\infty$$

Remark

The index p defines the Minkowski norm used to measure the approximation error. The index α is the decay order of the approximation error. Spaces $\mathcal{A}_q^\alpha(L_p)$ are spaces of functions such that the N–term L_p error decays like $N^{-\alpha}$. The index q is a decay scale subdivision index which is of secondary importance. We can note that $\mathcal{A}_{+\infty}^\alpha(L_2) = \mathcal{A}^\alpha$.

A function f is in some nonlinear approximation space if its expansion over the basis $\mathcal{G}$ is sparse, which means that the sequence of coefficients rearranged in decreasing order fulfills

$$|\langle f, g_{n_k}\rangle| \leq Bn^{-\alpha} \tag{8.1}$$

for a large α. An almost equivalent definition of this property is

$$\sum_{n\in\mathbb{N}} |\langle g, g_n\rangle|^{1/\alpha} < +\infty$$

We can then prove that the nonlinear N term approximation error decays like $N^{1/2-\alpha}$ and that therefore $f \in \mathcal{A}^{\alpha-1/2}$.

Nonlinear approximation is used for example to truncate expansions of signals or pictures over wavelet bases. The nonlinear approximation spaces are then Besov spaces, that are larger than Sobolev spaces. A Besov space is mainly defined with two indices. The first index α is an overall regularity index while the second one p is a tolerance index allowing some sparse singularities. A uniformly α–Lipschitz function is in the Besov space $B^\alpha(L_p)$. If now this function has a finite number of singularities where it is only $(\alpha - d/p)$–Lipschitz, and is uniformly α–Lipschitz elsewhere, it still is in this same Besov space.

The definition of a Besov space relies on the smoothness modulus of a function

Definition 8.4 (Smoothness modulus)

We define the finite difference operator of order r and step $\boldsymbol{h}$ $\Delta_{\boldsymbol{h}}^r$ to be

$$\Delta_{\boldsymbol{h}}^1(f)(\boldsymbol{x}) = f(\boldsymbol{x}+\boldsymbol{h}) - f(\boldsymbol{x}) \qquad \text{for } r = 1$$

and with the recursive relationship

$$\Delta_{\boldsymbol{h}}^r(\Delta_{\boldsymbol{h}}^{r'}(f)) = \Delta_{\boldsymbol{h}}^{r+r'}(f)$$

We can easily show that the operator $\Delta_{\boldsymbol{h}}^r$ can be written as

$$\Delta_{\boldsymbol{h}}^r(f)(\boldsymbol{x}) = \sum_{k=0}^{r} (-1)^{r-k} C_r^k f(\boldsymbol{x}+k\boldsymbol{h})$$

We then define the smoothness modulus of order r and step size h as:

$$\omega_r(f,h) = \sup_{|\boldsymbol{h}'|\leq h} \|\Delta_{\boldsymbol{h}'}^r(f)\|_p$$

We see that this smoothness modulus decreases with the step size h, and goes to 0 as $h \to 0$ if the function f is continuous. If f is r times derivable, the smoothness modulus decreases in h^r.

The choice of an exponent p will allow some singularities on f. If we suppose that p is smaller than $1/r$, that f is compactly supported, r times derivable on $\mathbb{R}^*$ (of bounded r^{th} derivative) and discontinuous on 0, the $r+1$ order finite difference of step h will be uniformly bounded on $\mathbb{R} - [-rh, rh]$ by some multiple of h^r, and bounded by a constant on the interval $[-rh, rh]$.

In this case,

$$\int_{\mathbb{R}} |\Delta_{\boldsymbol{h}}^r(f)(x)|^p dx \leq M|\boldsymbol{h}|^{pr} + M'|\boldsymbol{h}| \qquad \leq M'|\boldsymbol{h}|^{pr} \qquad \text{for a small enough } |\boldsymbol{h}|$$

which implies

$$\omega_r(f,h)_p \leq M'' h^r$$

For small exponents p the smoothness modulus "oversees" some singularities. More generally, assume that f is uniformly α–Lipschitz on its domain except on the support S of some singularity where it is only β–Lipschitz. Also assume that S has a Hausdorff dimension δ and a Hausdorff measure $m_\delta(S) < +\infty$. Then,

$$\begin{aligned} \int |\omega_h^\alpha(f)(x)|^p dx &\leq M h^{\alpha p} + m_d(\mathcal{S}) h^{d-\delta} h^{\beta p} \\ &\leq M' h^{\alpha p} \qquad \text{if } p < (d-\delta)/(\alpha-\beta) \end{aligned}$$

The smoothness modulus ignores singularities that can be stronger (i.e with large $\alpha - \beta$) and of support S of higher Hausdorff dimension when p decreases. We can now define:

Definition 8.5 (Besov space)

A function f is in the Besov space of indices α, p and q denoted $B_q^\alpha(L_p)$ if

$$\begin{aligned} \int_0^{+\infty} \left(h^{-\alpha}\omega_{\lfloor\alpha+1\rfloor}(f,h)_p\right)^q \frac{dh}{h} &< +\infty \qquad \text{if } q < +\infty \\ \sup_{h>0} \left(h^{-\alpha}\omega_{\lfloor\alpha+1\rfloor}(f,h)_p\right) &< +\infty \qquad \text{if } q = +\infty \end{aligned}$$

In the same way as for the definition of nonlinear approximation spaces, the index q is of secondary importance with comparison to α and p.

Nonlinear approximation theory says that nonlinear approximation spaces for wavelet bases are Besov spaces. This is exactly stated as

Theorem 8.1

Let $\psi_{j\boldsymbol{k}}^s$ be a dyadic orthonormal wavelet basis of $L_2(\mathbb{R}^d)$, i.e. fulfilling $\psi_{j\boldsymbol{k}}^s = 2^{dj/2}\psi^s(2^j\boldsymbol{x} - \boldsymbol{k})$, then

$$\mathcal{A}_\tau^{\alpha/d}(L_p) = B_\tau^\alpha(L_\tau)$$

if

$$\tau = \frac{1}{s/d + 1/p}$$

provided that the wavelets we use have at least $\lfloor\alpha+1\rfloor$ vanishing moments and are in the same Besov space. Under these provisions, nonlinear approximation spaces in wavelet bases are Besov spaces.

8.1.3 Additional structure and tree approximation

Since we have a limited amount of information on the function we want to estimate (a finite number of samples), several different interpolations corresponding to several choices of wavelet basis subfamily might provide a stable linear system of interpolation constraints. An extreme example is when all wavelets are chosen to be of very high resolution, and our estimate is then some finite sum of very localized peaks, which is usually called overfitting. To go around this, we have to use some additional a priori information on some order of preference according to which we choose the wavelets of our subfamily. We do this by imposing an additional structure on this subfamily.

Moreover, using the results of nonlinear approximation in its most general setting, that is looking for an interpolant as an expansion over any kind of wavelet subfamily is not possible for at least two reasons. The first reason is a storage problem. In nonlinear approximation, we not only have to code the selected coefficient values:

$$\mathcal{C}_N = \{N, g_1, \dots g_N\}$$

but also the corresponding wavelet indices. The corresponding code is therefore:

$$\mathcal{C}_N^{nl} = \{N, n_1, \dots, n_N, g_{n_1}, \dots, g_{n_N}\}$$

in order to rebuild the signal approximation.

Such a code can in theory be of arbitrarily high size if the indices can be chosen regardless of their resolution.

The second reason is algorithmic complexity. The set

$$\mathcal{C}d = \mathcal{G} - \{\psi_{j\boldsymbol{k}} : (j, \boldsymbol{k}) \notin I\}$$

of candidate wavelets among which we look for an additional wavelet $\psi_{j\boldsymbol{k}}$ to increase the subfamily of wavelets over which we expand our approximation to get $I \cup \{\psi_{j\boldsymbol{k}}\}$ is in this case infinite. To limit ourselves to a finite set of candidate functions, we have to rely on more structured representations than *any* subfamily:

$$f_N(x) = \sum_{n \in I_N} c_n g_n(x)$$

An often used strategy in image compression consists in using subfamilies that make up a tree. This choice is mainly suggested by heuristic considerations. High wavelet coefficients are usually the coefficients of wavelets whose support overlaps a singularity. Such wavelets make up a cone of influence of a singularity that can be arranged into a tree, as shown in figure 8.1.

This implies that we can consider that there is some strong statistical relationship between wavelet coefficients of close resolution j and location $2^{-j}\boldsymbol{k}$. When the wavelet basis is build from a set of S mother wavelets $(\psi^s)_{s=1\dots S}$ with the following definitions:

$$\psi^s_{j\boldsymbol{k}}(\boldsymbol{x}) = 2^d j/2 \psi^s(2^j \boldsymbol{x} - \boldsymbol{k}) \qquad\qquad s \in \{1 \dots S\},\ j \in \mathbb{Z},\ \boldsymbol{k} \in \mathbb{Z}^d$$

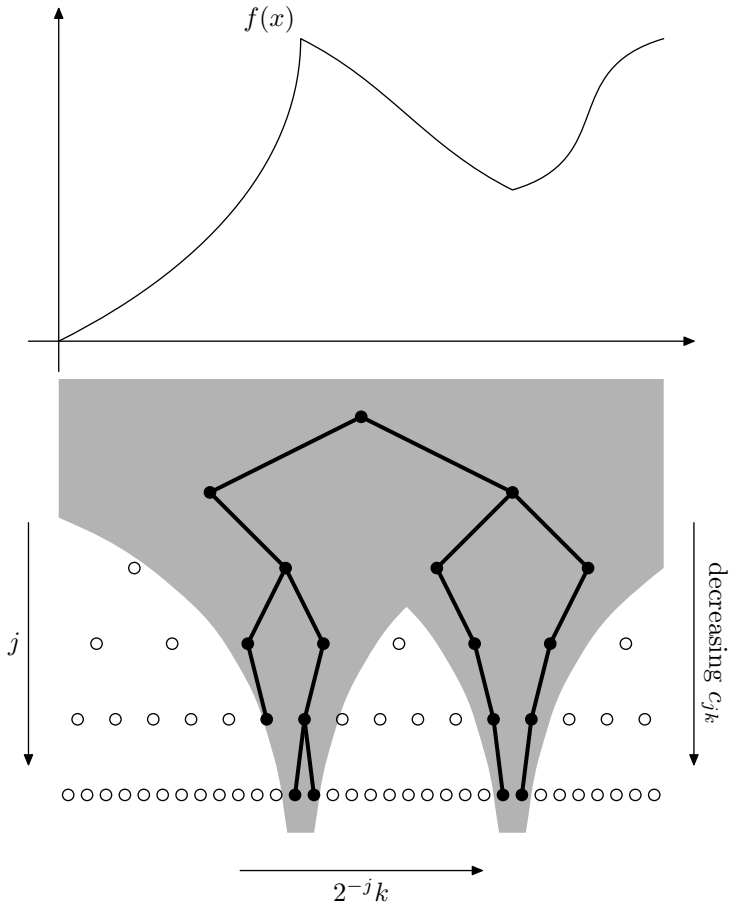


Figure 8.1: Cone of influence of two pointwise singularities, and corresponding subtree.

it is usual to choose as a tree structure the following relationship:

$$(j, \boldsymbol{k}, s)\, \mathcal{R}\, (j', \boldsymbol{k}', s') \Leftrightarrow \begin{cases} s = s' \\ j' = j + 1 \\ k'_i \in \{2k_i, 2k_i + 1\} \end{cases}$$

In this case, the set of candidate functions to increase a subfamily of already chosen wavelets is finite:

$$\mathcal{C}d = \{\psi^s_{j\boldsymbol{k}} : (j, \boldsymbol{k}, s) \notin I,\ \exists (j', \boldsymbol{k}', s') \in I \text{ such that } (j', \boldsymbol{k}', s')\mathcal{R}(j, \boldsymbol{k}, s)\}$$

If we restrict ourselves to choosing tree structured subfamilies of wavelets, i.e. to use *tree approximations*, we can fear that the resulting error decay as the number of wavelets we use increases is slower than for the most general setting of nonlinear approximation. Cohen, DeVore *et al.* have proved that in fact, the loss of approximation capacity caused by this restriction is small. Making a smoothness assumption on the approximated function with smoothness indices that are arbitrarily close to the original ones is sufficient to obtain with tree approximation the same error decay as with general nonlinear approximation.

A theorem of Cohen, DeVore, *et al.* [DeV98b, CDDD98] says that tree approximation spaces are close to nonlinear approximation spaces. More precisely, if denote by $\mathcal{T}^\alpha_q(L_p)$ the tree approximation space defined in a manner similar to $\mathcal{A}^\alpha_q(L_p)$ (where now the only subexpansions we can choose have to make up trees), we have

$$\mathcal{T}^{\alpha/d}_{+\infty}(L_p) \supset B^\alpha_q(L_\mu)$$

as soon as

$$\mu > \tau = \frac{1}{\alpha/d + 1/p}$$

provided that the wavelets we use have $\lfloor \alpha \rfloor + 1$ vanishing moments and lie in the Besov space $B^\alpha_q(L_\mu)$.

The advantage of tree representation in terms of coding size is immediate. Coding the selected coefficients now only requires $\mathcal{O}(N)$ bits where N is the number of wavelet coefficients. Indeed, it now only suffices to say for each wavelet which of its sons also are in the wavelet subfamily.

Last but not least, we can also propose an interpretation of this strategy in the light of the recent work of Vapnik on structural risk minimization. In 7.7.3, we have seen a framework to implement Vapnik's structural risk minimization principle. This framework consists in building a increasing sequence $(S_n)_{n\in\mathbb{N}}$ of approximating function families with increasing VC dimensions. Minimizing the structural risk over this sequence can be done by looking in a finite number of these subspaces S_n for the best trade-off between regression error in S_n and the generalization risk related to the VC dimension of S_n.

Tree approximation can then be interpreted the following way : we denote by $A \in \mathcal{A}$ the wavelet subfamilies that make up trees. The VC dimension of the subspace span A spanned by such a family is either card A or card $A + 1$. These subspaces can replace the increasing sequence of S_n sets. The main difference between these approaches is that in the former one, we have a sequence of families $(S_n)_{n\in\mathbb{N}}$ indexed by a totally order set of indices, while in the latter one, the index ordering of $(\text{span } A)_{A\in\mathcal{A}}$ is only partial, as is displayed in Fig. 8.2.

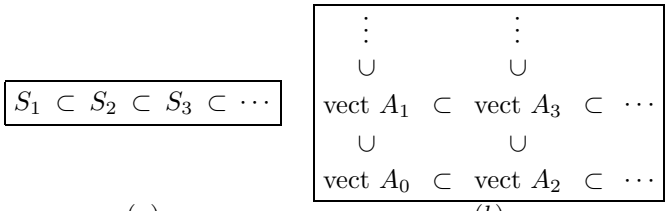


Figure 8.2: Approximation families for structural risk minimization. In (a), the increasing sequence suggested by Vapnik. In (b), another choice structure corresponding to subtree approximations.

8.1.4 Limits of wavelet approximation

When the dimension d of the underlying domain of approximation increases, singularities of smooth support and of dimension higher than 0 can be expansive to code with wavelets, as is shown in Fig. 8.3.

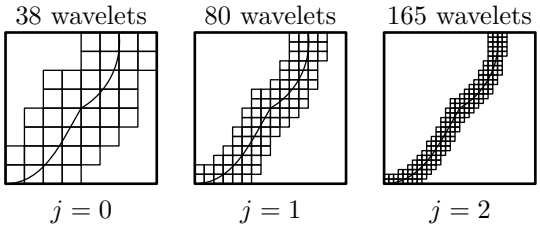


Figure 8.3: Cost of representing a singularity. The number of coefficients whose value is affected by the presence of the singularity increases like $2^{\delta j}$ where δ is the Hausdorff dimension of the singularity support.

Singularities of high Hausdorff dimension

Let us consider a piecewise constant function:

$$f(\boldsymbol{x}) = 1_{(\boldsymbol{x} \in \Omega)}$$

where the boundary $\partial\Omega$ is of finite length and bounded curvature. This function is of bounded variations. The wavelets of resolution j that overlap the singularity are $\mathcal{O}(2^{(d-1)j})$ and their coefficient is bounded by some $\mathcal{O}(2^{-dj/2})$. The corresponding L_2 energy is therefore of magnitude order

$$\begin{aligned} \sigma_j(f) &= \sum_{n, \boldsymbol{k} \in \mathbb{Z}^d} \left[c^n_{j\boldsymbol{k}}(f)\right]^2 \\ &\leq M 2^{-j} \end{aligned}$$

An approximation on the first j resolutions will contain

$$1 + \cdots + 2^{(d-1)(j-1)} + 2^{(d-1)j} \propto 2^{(d-1)j}$$

terms, and the corresponding approximation error will be

$$\|f - \sum_{j'<j} \sum_{n,\boldsymbol{k}} c^n_{j'\boldsymbol{k}} \psi^s_{j'\boldsymbol{k}}\|_2 = \mathcal{O}(2^{-j/2})$$

We thus get an error estimate for $N = 2^{(d-1)j}$ terms of

$$\epsilon_N(f) = \mathcal{O}(N^{-\frac{1}{2(d-1)}}) \tag{8.2}$$

We can see that this estimate is the same as the one provided by theorem 8.1. Leaving the index q aside, we see that the function $1_{(\boldsymbol{x}\in\Omega)}$ is in any Besov space $B^\alpha(L_p)$ where $p \leq 1/\alpha$. The theorem then says that

$$\mathcal{A}^\alpha(L_2) = B^{d\alpha}(L_\tau)$$

where

$$\tau = \frac{1}{\alpha + 1/2}$$

This theorem applies whenever

$$\begin{aligned} & \tau \leq \frac{1}{d\alpha} \\ \Leftrightarrow \quad & \alpha + \frac{1}{2} \geq d\alpha \end{aligned}$$

thus for $\alpha \leq \frac{1}{2(d-1)}$, and we get the same error decay estimate as before in (8.2).

David Donoho and Emmanuel Candès have developed modified wavelet transforms (ridgelet transforms) that allow to represent hyperplane singularities in a sparser way. Unfortunately, these representations are then conversely less efficient for pointwise singularities. They can therefore be used to represent function as part of a redundant dictionary that will contain classical wavelets as well as ridgelets which share the task of representing well different types of singularities. A detailed description of ridgelets can be found in Donoho's papers and Emmanuel Candès' PhD thesis [Can97, Don98a, Can98].

In the case of the above function, some approximation schemes are more efficient than nonlinear approximation in wavelet bases. If we consider the case of a 2–dimensional domain, and the function f to approximate is $1_{(\boldsymbol{x}\in\Omega)}$ where the boundary of Ω is of finite length and bounded curvature, such a function can be approximated by sums of N triangle characteristic functions of triangles $(T_n)_{n=1\ldots N}$, with an L_2 approximation error of magnitude less than $\mathcal{O}(N^{-1})$, as shown by N. Dyn and S. Rippa [DLR90].

Such an approximation can thus provide a better approximation error for the same code than a wavelet approximation, by taking advantage of the smoothness of the boundary $\partial\Omega$.

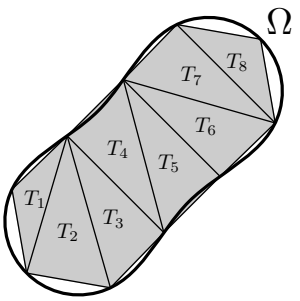


Figure 8.4: approximation of a characteristic function of smooth boundary domain with sums of T_n triangle characteristic functions.

This can also be used to show that similar approximation rates can be achieved with neural networks. Indeed, a triangle can be represented by three linear inequalities:

$$T_n = \{\boldsymbol{x} : \lambda_n^1(\boldsymbol{x}) \geq 0, \lambda_n^2(\boldsymbol{x}) \geq 0, \lambda_n^3(\boldsymbol{x}) \geq 0\}$$

If we choose as a transfer function the discontinuous sigmoid $\sigma(t) = 1_{(t>0)}$, a triangle characteristic function can be written as

$$1_{(\boldsymbol{x}\in T_n)} = \sigma\left(\sum_{k=1}^{3} \sigma(\lambda_n^k(\boldsymbol{x})) - 2,5\right)$$

and thus the function $1_{(\boldsymbol{x}\in\Omega)}$ can be approximated with an error of order N^{-1} with $4N+1$ neurons. The optimal approximation error with neural nets is thus at most of order $\mathcal{O}(N^{-1})$ where N is the number of neurons.

These two approximation methods are in this case more efficient than wavelet representations that only achieve a rate of $\mathcal{O}(N^{-1/2})$.

Very high dimensional input spaces

For very high dimensional input spaces, the wavelet representation of a function suffers from a severe curse of dimensionality. Indeed, the efficiency of a wavelet representation relies on a prediction of new function values on the basis of other ones. For a very simple case of monovariate wavelets that are piecewise linear, the prediction scheme predicts a third value on the basis of the two first ones. So we need at least two samples to be able to make a prediction. In dimensions d higher than 1, tensor product wavelets are piecewise d-linear wavelets, and a first prediction can only be made if we thus already have 2^d samples. If the dimension is high, we never have so many samples, so we cannot expect to have any sparse representation of a function in a wavelet basis. For these cases, the wavelet representation is not well suited, and ridge function of radial basis function expansion provide a feasible answer.

To summarize, we have seen that some tools can be more efficient than wavelets to represent piecewise regular functions. However, the algorithmic methods involved are then

complex. They either consist in looking in a redundant dictionary [Don98b] for an optimal approximation, or in manipulating neural networks, whose learning algorithms are barely provable, or in using adaptive meshes. In this last case, the drawback appears in high dimensions, where the complexity of the approach increases drastically.

Nevertheless, wavelets already allow to achieve a good trade-off, because they have good approximation properties and in the same time allow to build learning algorithms of reasonable complexity, and whose convergence can be proved, as we shall see soon. As said, all this will in practice hold for input spaces of reasonable dimension.

8.1.5 Interpolating wavelet trees

Using trees of interpolating wavelets for our purpose involves some technical issues we will review now.

The first issue is a problem related to the tree representation of multivariate wavelet expansions. If the dimension of the domain is higher than 1, we usual have several wavelet shapes corresponding to the mother wavelets. In general, we have $2^d - 1$ mother wavelets and thus $2^d - 1$ different wavelet shapes. The wavelets are spreading among as many disjoint trees that are handled separately.

It seemed artificial and expensive to us to choose between several disjoint wavelet trees. We therefore propose a new structure to put an interpolating wavelet basis where we only have a single wavelet shape.

We mimic for this purpose Donoho's construction of monovariate interpolating wavelet bases. If we are given an interpolating function ϕ, that thus fulfills

$$\phi(k) = \delta_{k0} \qquad \forall k \in \mathbb{Z}$$

or also

$$\sum_{k\in\mathbb{Z}} \hat{\phi}(\xi + 2k\pi) = 1$$

Donoho suggests to choose as a wavelet the function $\psi(x) = \phi(2x - 1)$, thus a function that has exactly the same shape as ϕ. The detail spaces W_j are built with a basis which is a subsampled basis of that of the scale space V_{j+1}. The classical way to build multivariate wavelets on the basis of monovariate ones consists then in setting

$$\psi^\epsilon(x) = \begin{cases} \phi(x) & \text{if } \epsilon = 0 \\ \psi(x) & \text{if } \epsilon = 1 \end{cases}$$

and in choosing the $2^d - 1$ mother wavelets as

$$\psi^{\boldsymbol{\epsilon}}(\boldsymbol{x}) = \psi^{\epsilon_1}(x_1) \times \cdots \times \psi^{\epsilon_d}(x_d) \qquad \text{where } \boldsymbol{\epsilon} \in \{0,1\}^d$$

If ϕ and ψ are a scaling function and a wavelet fulfilling $\psi(x) = \phi(2x - 1)$, the $2^d - 1$ multivariate mother wavelets all have different shapes. Another approach consists in applying Donoho's scale space subsampling method directly on the d–variate scale spaces.

We therefore set $\phi(\boldsymbol{x}) = \phi(x_1) \times \cdots \times \phi(x_n)$. We then define

$$V_j = \overline{\text{vect}\{\phi(. - \boldsymbol{k}) : \boldsymbol{k} \in \mathbb{Z}^d\}}$$
$$W_j = \overline{\text{vect}\{\phi(2. - \boldsymbol{k}) : \boldsymbol{k} \in \mathbb{Z}^d, \boldsymbol{k}/2 \notin \mathbb{Z}^d\}}$$

It can then easily be checked that

$$V_j \oplus W_j = V_{j+1}$$

and the dual basis of

$$\mathcal{B} = \{\phi_{j\boldsymbol{k}} : j = 0 \text{ and } \boldsymbol{k} \in \mathbb{Z}^d\} \cup \{\phi_{j\boldsymbol{k}} : j > 0 \text{ and } \boldsymbol{k} \in \mathbb{Z}^d - 2\mathbb{Z}^d\}$$

is then

$$\tilde{\phi}_{0\boldsymbol{k}}(\boldsymbol{x}) = \delta(\boldsymbol{x} - \boldsymbol{k}) \qquad \boldsymbol{k} \in \mathbb{Z}^d$$
$$\tilde{\phi}_{j\boldsymbol{k}}(\boldsymbol{x}) = \delta(\boldsymbol{x} - 2^{-j}\boldsymbol{k}) - \sum_{\boldsymbol{k}' \in \mathbb{Z}^d} \phi(\boldsymbol{k}/2 - \boldsymbol{k}')\delta(\boldsymbol{x} - 2^{-j+1}\boldsymbol{k}') \qquad j > 0,\ \boldsymbol{k}/2 \notin \mathbb{Z}^d$$

The dual wavelets are finite linear combinations of Dirac masses, and are not separable. Since however we never use then explicitly, this complexity does not impair our approach. The simplicity of the structure of wavelet basis however will be useful in our approach.

Tree structures

We can easily give to such a wavelet family a tree structure when $d = 1$. A wavelet is indexed by two indices j and k. The tree structure is then the following:

$$(j,k) \text{ has the sons } \begin{cases} (j+1, 2k-1) \\ (j+1, 2k+1) \end{cases}$$

In higher dimensions however, it is not possible to find such a simple structure again, unless we add some redundancy to the tree structure. The nodes in the tree are indexed by pairs $(j, \boldsymbol{k})$ where $j \in \mathbb{N}$ and $\boldsymbol{k} \in \mathbb{Z}^d$. The tree structure is then for example:

$$(j, \boldsymbol{k}) \text{ has the sons } (j+1, 2\boldsymbol{k} + \boldsymbol{\epsilon}) \qquad \boldsymbol{\epsilon} \in \{0,1\}^d$$

Some nodes then do not correspond to a wavelet coefficient: those for which $j > 0$ and $\boldsymbol{k} \in 2\mathbb{Z}^d$.

Is a tree structure always necessary?

We will see that for some interpolation schemes, we can give up the tree structure for the subfamily over which we expand our function estimate.

One of the main advantages of the tree structure is to give a way to limit the number of candidate functions. In the approach we will describe next, we will see that we can partly relax this constraint when using a so called *allocation* method. In some cases, this allocation method will consist in doing a tree descent, and in other cases not.

8.1.6 Learning and wavelet networks

We know two contributions on learning with wavelet networks we will detail a little now. The first one has been made by Zhang and Benveniste [ZB92] and later Zhang [SZL+95, Zha97]. The second one has been written by Pati and Krishnaprasad [PK93]. Other approaches have been described, but we will not review them here, because they are too far from ours.

The simplest method is that suggested by Pati and Krishnaprasad. They suggest to learn an unknown function by looking for its expansion over a finite subfamily of a wavelet basis. They select a subfamily, either by spectral analysis of the samples, or by progressive refinement, and solve for the wavelet coefficients by inverting a regression system, to get the $c_{j\boldsymbol{k}}$ coefficients in the expansion

$$f = \sum_{j,\boldsymbol{k}} c_{j\boldsymbol{k}} \psi_{j\boldsymbol{k}}$$

To show the similarity with classical neural networks, they describe an iterative gradient descent method to learn the expansion coefficients from individual computations considering the input samples one at a time. (Note that the error functional is quadratic in the unknown parameters).

The approach suggested initially by Zhang and Benveniste consists in looking for a function approximation of the form:

$$f = \sum_{k} a_k \psi(B_k \boldsymbol{x} - c_k)$$

where the matrices B_k and the vectors c_k are to be modified iteratively by the learning algorithm. Because the dependence of the solution w.r.t the parameters is strongly nonlinear, little can be said about the convergence of such gradient descent methods.

Then, Zhang developed a learning algorithm that consists in selecting wavelets from a wavelet frame, which is so far very close to what we are intending to do, and then to solve the resulting regression system to find the expansion coefficients in

$$f = \sum_{(j,\boldsymbol{k}) \in I} c_{j\boldsymbol{k}} \psi_{j\boldsymbol{k}}$$

His wavelet selection method is a two-pass method. A selection pass and selection/elimination pass.

The first selection pass consist in choosing all the wavelet whose support overlaps some measure point $\boldsymbol{x}$. This way, he limits his considerations to wavelets that are likely to explain the measures he has.

The second pass can be done in two opposite ways. Either by selecting from this subset wavelet to make the finite wavelet subexpansion, or selecting out wavelets that are least useful for representing accurately the learned function. The first possibility consists in finding iteratively the wavelet to add that explains best the regression residual error. This is done by an orthonormalization of the wavelet in an adapted metric:

$$\langle f, g \rangle = \sum_{n=1}^{N} f(\boldsymbol{x}_n) \overline{g(\boldsymbol{x}_n)}$$

The second possibility consists in crossing out wavelets of a full expansion $\sum_{(j,\boldsymbol{k})\in I} c_{j\boldsymbol{k}}\psi_{j\boldsymbol{k}}$ on the basis of the current corresponding wavelet coefficient. If this coefficient is beneath a given threshold, this means that the wavelet is not helpful in representing the function, and it is crossed out from the subset I. The system is resolved to readjust the coefficients.

In either case, the choice of the ideal number of wavelets is based on *generalized cross validation criterion*. This consists in minimizing the following composite functional

$$GCV = \frac{1}{N}\sum_{n=1}^{N}\left(f(\boldsymbol{x}_k) - y_k\right)^2 + 2\frac{\operatorname{card} I}{N}\sigma^2$$

where I is the wavelet index set we use, and thus $\operatorname{card} I$ the number of these wavelets. This functional is very similar to other ones, in that it is the sum of an empirical regression error, and of a capacity control term, that is increasing with the number of wavelets in the expansion. σ is the estimated standard deviation of the measuring noise.

8.2 Interpolating wavelets and approximation grids

We suppose we have a wavelet family we write as

$$\mathcal{B} = (\phi_{j\boldsymbol{k}})_{j\in\mathbb{N},\boldsymbol{k}\in K_j}$$

mapping elements of a starting space D to any normed vector space.

In these notations, j is the resolution index and $\boldsymbol{k}$ is a translation index. If ϕ is a Deslauriers–Dubuc scaling function, we can for instance define

$$\begin{aligned}
\phi_{j\boldsymbol{k}}(\boldsymbol{x}) &= \phi(2^j\boldsymbol{x} - \boldsymbol{k}) \\
K_0 &= \mathbb{Z}^d \\
K_j &= \{\boldsymbol{k}\in\mathbb{Z}^d : \boldsymbol{k}/2 \notin \mathbb{Z}^d\} \qquad\qquad j \geq 1
\end{aligned}$$

and obtain this way a basis of functions defined over all $\mathbb{R}^d$. In a hypercube, we will have to take $\phi_{j\boldsymbol{k}}$ to be a scaling function obtained by dyadic Deslauriers–Dubuc refinement adapted to the interval. The sets of indices K_j are then defined as

$$\begin{aligned}
K_0 &= \{0,1\}^d \\
K_j &= \left\{\boldsymbol{k}\in\{0,\ldots,2^j\}^d : \boldsymbol{k}/2 \notin \mathbb{Z}^d\right\} \qquad\qquad j \geq 1
\end{aligned}$$

In either case, we denote the whole set of $(j,\boldsymbol{k})$ index pairs that describes all wavelets in the basis by

$$K = \bigcup_{j=1}^{+\infty}\{j\}\times K_j$$

We denote $\nu_{j\boldsymbol{k}} \in \mathbb{R}^d$ the center of the wavelet of indices $(j,\boldsymbol{k})$ which is for the above example

$$\nu_{j\boldsymbol{k}} = 2^{-j}\boldsymbol{k}$$

We suppose that the wavelet basis $\mathcal{B}$ is an interpolating wavelet basis. This means that

$$\begin{aligned} \phi_{j\boldsymbol{k}}(\nu_{j\boldsymbol{k}'}) &= \delta_{\boldsymbol{k}\boldsymbol{k}'} && \forall\, \boldsymbol{k},\, \boldsymbol{k}' \\ \phi_{j'\boldsymbol{k}'}(\nu_{j\boldsymbol{k}}) &= 0 && \forall\, j' > j,\, \boldsymbol{k},\, \boldsymbol{k}' \end{aligned}$$

For each resolution j, we define a partition of D into attraction basins of the wavelets $\phi_{j\boldsymbol{k}}$:

$$D = \bigcup_{\boldsymbol{k} \in K_j} B_{j\boldsymbol{k}}$$

where $\nu_{jk} \in \overset{\circ}{B}_{jk}$. In practice, the attraction basin is a Voronoï cell, i.e.

$$B_{j\boldsymbol{k}} = \{x \in D : |x - \nu_{j\boldsymbol{k}}| \leq |x - \nu_{j\boldsymbol{k}'}| \forall \boldsymbol{k}' \in K_j\}$$

up to a subset of measure 0.

If we use triadic interpolation wavelets, for which a scaling function of resolution j and translation index $\boldsymbol{k}$ is written as $\phi_{j\boldsymbol{k}}(x) = \phi(3^j \boldsymbol{x} - \boldsymbol{k})$, similar definitions can be given. Namely

$$\nu_{j\boldsymbol{k}} = 3^{-j}\boldsymbol{k}$$

and for $j \geq 1$

$$K_j = \begin{cases} \{\boldsymbol{k} \in \mathbb{Z}^d : \boldsymbol{k}/3 \notin \mathbb{Z}^d\} & \text{if } D = \mathbb{R}^d \\ \{\boldsymbol{k} \in \{0 \dots 3^j\}^d : \boldsymbol{k}/3 \notin \mathbb{Z}^d\} & \text{if } D = [0,1]^d \end{cases}$$

Whatever the scale of resolutions we use (either triadic or dyadic), we can notice that a rational dyadic (resp. triadic) vector which can be written as $2^{-j}\boldsymbol{k}$ (resp. $3^{-j}\boldsymbol{k}$) is the center of a unique wavelet in the basis $\mathcal{B}$. The true indices $(j', \boldsymbol{k}')$ of this wavelet are uniquely defined by the following properties:

$$2^{-j}\boldsymbol{k} = 2^{-j'}\boldsymbol{k}' \quad (\text{resp. } 3^{-j}\boldsymbol{k} = 3^{-j'}\boldsymbol{k}')$$
$$j = 0 \text{ or else } (j > 0 \text{ and } \boldsymbol{k} \notin 2\mathbb{Z}^d \text{ (resp. } \boldsymbol{k} \notin 3\mathbb{Z}^d))$$

8.3 Allocation scheme

In this section, we describe a scheme aimed at choosing a subfamily of wavelets from a basis, based on the set of measure points we have. This scheme is designed so as to satisfy a number of optimality criteria we detail thereafter.

The purpose of these allocation schemes is to choose a wavelet subfamily to build an interpolating function on the basis of a set of pointwise measures. Two properties are necessary:

- the scheme must build a subfamily in which the interpolation constraints provide a function that is a priori smooth. This implies that low resolution wavelets should be preferred to higher resolution wavelets;

- fine resolution wavelet coefficients should only appear where the density of measure points is high enough to provide the corresponding high resolution information. We thus expect from this scheme that it selects high resolution wavelets only where the density of measure is sufficiently high.

We suppose we are given a sequence of measures $(x_n, y_n)_{n=1\dots N}$. In practice, the allocation scheme will only work on the basis of the location of the measure points x_n, and not of the measured values. In what follows, $\mathcal{X}$ will denote the set of measure points $\{x_n : n = 1 \dots N\}$.

Definition 8.6 (Admissible allocation)

An admissible allocation of a finite set of measure points $\mathcal{X}$ into a wavelet basis is a one to one mapping i from $\mathcal{X}$ to K such that any measure $\boldsymbol{x}$ allocated to a wavelet $\phi_{i(\boldsymbol{x})}$ is in the attraction basin of this wavelet

$$x \in B_{i(x)} \qquad \text{for all } x \in \mathcal{X}.$$

and that a measure is allocated to a wavelet if it cannot be allocated to a coarser scale wavelet, namely:

$$i(x) = (j, \boldsymbol{k}),\ x \in B_{j'\boldsymbol{k}'} \text{ with } j' < j \Rightarrow \exists x' \in \mathcal{X} \text{ such that } i(x') = (j', \boldsymbol{k}')$$

We denote by I the range of i, which indexes the chosen subfamily of selected wavelets.

In general, (as soon as wavelets of different scales are in I), several different admissible allocations are possible. We only consider such allocations that are "best" according to an ordering we can define between two allocations of a same set of measure points $\mathcal{X}$.

Definition 8.7

*An allocation i of set of measure points $\mathcal{X}$ into a wavelet basis is **superior** to another allocation i' if and only if there exists some $j_0 \in \mathbb{N}$ such that the following two conditions (C_1–C_1') and (C_2) are fulfilled:*

1. *For all j such that $0 \leq j < j_0$, we have*

$$I \cap (\{j\} \times K_j) = I' \cap (\{j\} \times K_j) \qquad (C_1)$$

$$(j, \boldsymbol{k}) \in I \Rightarrow {i}^{-1}(j, \boldsymbol{k}) = {i'}^{-1}(j, \boldsymbol{k}) \qquad (C_1')$$

   which means that the allocations i and i' are the same up to the resolution index $j_0 - 1$.

2. *Since these allocations are both admissible, the wavelets selected at scale j_0 are also the same:*

$$I \cap (\{j_0\} \times K_{j_0}) = I' \cap (\{j_0\} \times K_{j_0})$$

   The second condition is then that the measure allocated at scale j_0 be closer to the corresponding wavelet centers for i than for i', i.e.

$$(j_0, \boldsymbol{k}) \in I \Rightarrow |{i}^{-1}(j_0, \boldsymbol{k}) - \nu_{j_0\boldsymbol{k}}| \leq |{i'}^{-1}(j_0, \boldsymbol{k}) - \nu_{j_0\boldsymbol{k}}| \qquad (C_2)$$

We will see that this allocation scheme has two good properties. One can show a uniqueness result, and one can show that the allocation converges (when the number of measure points increases) to an exact correspondence with the wavelet centers. As a result, the corresponding subsystems of linear constraints are then invertible and well conditioned.

8.3.1 Quasi-uniqueness and existence of an optimum

Strictly speaking, there can be cases where there is no absolutely optimal allocation. This will mean that there exist two equivalent allocations.

Definition 8.8 (Equivalent allocations)

One says that two allocations are equivalent if there exists a resolution j_0 such that conditions (C_1–C_1') and (C_2) above are fulfilled, where the inequalities in (C_2) are all equalities, but the allocations differ at scale j_0 :

$$(j_0, \boldsymbol{k}) \in I \Rightarrow |i^{-1}(j_0, \boldsymbol{k}) - \nu_{j_0 \boldsymbol{k}}| = |{i'}^{-1}(j_0, \boldsymbol{k}) - \nu_{j_0 \boldsymbol{k}}| \qquad (C_2')$$

$$\exists \boldsymbol{k} \text{ such that } (j_0, \boldsymbol{k}) \in I \text{ and } i^{-1}(j_0, \boldsymbol{k}) \neq {i'}^{-1}(j_0, \boldsymbol{k}) \qquad (C_2'')$$

We can state the following lemma:

Theorem 8.2

Let $\mathcal{X}$ be a set of measure points. Then at least one of the following sentences is true:

- *there exists an optimal allocation, which is strictly better than any other admissible allocation.*
- *there exist two equivalent allocations.*

Proof. We will only give some milestones of the proof which is not especially technical. It consists in doing the allocation resolution by resolution the following way. We start by setting $j = 0$ and $\mathcal{X}_0 = \mathcal{X}$,

1. We name Λ_j the set of the $\boldsymbol{k} \in K_j$ such that $\mathcal{X}_j \cap B_{j\boldsymbol{k}} \neq \emptyset$. For all $\boldsymbol{k} \in \Lambda_j$, we denote by $x_{j\boldsymbol{k}}$ the measure point of $\mathcal{X}_j$ in the attraction basin B_{jk} which is closest to $\nu_{j\boldsymbol{k}}$. Since the attraction basins are not overlapping, the $x_{j\boldsymbol{k}}$ are all distinct. We can then set $i(x_{j\boldsymbol{k}}) = (j, \boldsymbol{k})$.

2. We define the set of measure points to allocate from the next resolution:

$$\mathcal{X}_{j+1} = \mathcal{X}_j - i^{-1}(\{j\} \times \Lambda_j)$$

The remaining measure points all fulfill

$$x \in \mathcal{X}_{j+1} \text{ and } x \in B_{j'\boldsymbol{k}} \text{ for } j' \leq j \Rightarrow \boldsymbol{k} \in \Lambda_{j'}$$

This process is iterated for $j = 1, 2, \ldots$ until $\mathcal{X}_j = \emptyset$. Since

$$\operatorname{card} \mathcal{X}_{j+1} < \operatorname{card} \mathcal{X}_j,$$

this algorithm finishes (after at most $j = \operatorname{card} \mathcal{X}$ steps). It is obvious that we obtain this way an optimal allocation, unless the definition of the $x_{j\boldsymbol{k}}$ is at some point not unique, which means that at a given resolution j_0, two measure points are at the same distance from a center $\nu_{j_0\boldsymbol{k}}$ in which case we have two equivalent allocations. □

8.3.2 Iterative allocation scheme

The scheme we describe below (of iterative allocation) allows to achieve an optimal allocation whenever it exists.

If the measure points $\boldsymbol{x}_n$ are realizations of a random variable X of bounded and non vanishing density, we can then prove that the situations such that there exists two equivalent allocations almost never happen.

This will guarantee that in the general setting (up to almost never happening cases), we always have a unique allocation, and that the result of the iterative allocation process is unique, and does not depend on the ordering of the measure points (x_n), but solely on the set of measure points $\mathcal{X}$.

Definition 8.9 (Iterative allocation scheme)

We suppose that we already have an optimal allocation i_n of a set of n points $\mathcal{X}_n$ into the wavelet basis. The allocation of the $n+1^{th}$ measure is done the following way:

Computing i_{n+1} from i_n *consists in allocating $\boldsymbol{x}$ from the scale $j = 0$.*

Allocating $\boldsymbol{x}$ from the scale j *consists in doing the following*

1. *Let $\boldsymbol{k} \in K_j$ such that $x \in B_{j\boldsymbol{k}}$.*
   (a) *Other measures already allocated at the same scale to other wavelets are not moved:*
   $$i_n(\boldsymbol{x}) = (j, \boldsymbol{k}') \text{with } \boldsymbol{k}' \neq \boldsymbol{k} \Rightarrow i_{n+1}(\boldsymbol{x}) = i_n(\boldsymbol{x})$$
   (b) *If no measure was allocated to $(j, \boldsymbol{k})$, i.e. $(j, \boldsymbol{k}) \notin I_n$, we set*
   $$i_{n+1}(\boldsymbol{x}) = (j, \boldsymbol{k})$$
   and i_{n+1} coincides with i_n for all remaining wavelets, i.e.
   $$i_n(\boldsymbol{x}) = (j', \boldsymbol{k}') \text{with } j' > j \Rightarrow i_{n+1}(\boldsymbol{x}) = i_n(\boldsymbol{x})$$
   The algorithm is finished.
   (c) *If a measure $\boldsymbol{x}'$ is already allocated to $(j, \boldsymbol{k})$, i.e. if $(j, \boldsymbol{k}) = i_n(\boldsymbol{x}')$, we say that there is a competition between the measures $\boldsymbol{x}$ and $\boldsymbol{x}'$. We denote by $\boldsymbol{x}_c$ the one which is closer to $\nu_{j\boldsymbol{k}}$ and $\boldsymbol{x}_f$ the one which is further. We then set*
   $$i_{n+1}(\boldsymbol{x}_c) = (j, \boldsymbol{k})$$
   and we allocate $\boldsymbol{x}_e$ from the scale $j + 1$.

This algorithm finishes in at most $j+1$ steps if j is the largest resolution in the set I_n of already selected wavelets, i.e.

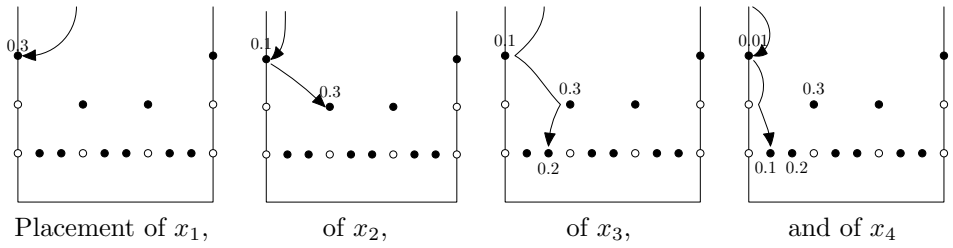
$$j = \max_{(j',\boldsymbol{k}')\in I_n} j'$$

We can prove recursively that we obtain in this way an optimal allocation (if there are no equivalent allocations). This most notably guarantees that the result of an iterative allocation is always the same for a same set of measure points $\mathcal{X}$, and does not depend on the measure point ordering.

Allocation example

We give here an example of iterative allocation in the case of triadic interpolating wavelets. The allocated measures are respectively

$$x_1 = 0,3 \qquad x_2 = 0,1$$
$$x_3 = 0,2 \qquad x_4 = 0,01$$



Placement of x_1, of x_2, of x_3, and of x_4

Figure 8.5: Example of an iterative allocation

The result of this allocation is represented in figure 8.5. Solid circles represent tree nodes that correspond to a basis function, while the others do not, and are placed here to maintain a tree structure. The measure x_1 is allocated to the wavelet ϕ_{00}. Then, measure x_2 being closer to ν_{00} than x_1, x_1 is pushed down one scale to be allocated to ϕ_{11}, etc.

8.3.3 Asymptotic behavior

We will now prove a theorem that strongly motivates this approach. If we iteratively place measure points and if the x_n are i.i.d. realizations of a random variable X of strictly positive (and bounded) density, the point allocated to a fixed wavelet converges almost surely to the center of this wavelet.

Theorem 8.3

We assume that the measure points $\boldsymbol{x}_n$ are i.i.d. realizations of a random variable X of strictly positive and bounded density. We note i_n the allocation obtained by iterative allocation of the measures $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ into the wavelet tree and I_n the set of indices of the selected wavelets (the range of i_n).

We can then prove that for all fixed $j \geq 0, \boldsymbol{k} \in K_j$,

$$\exists n_0 \text{ such that } n \geq n_0 \Rightarrow (j, \boldsymbol{k}) \in I_n$$
$$\boldsymbol{x}_{i_n^{-1}(j,\boldsymbol{k})} \to \nu_{j\boldsymbol{k}} \quad n \to +\infty$$

almost surely.

Proof. We choose a step $\epsilon < 2^{-j-1}$. We denote by $b_\epsilon(j, \boldsymbol{k})$ the set

$$b_\epsilon(j, \boldsymbol{k}) = \{x \in D : |x - \nu_{j\boldsymbol{k}}| < \epsilon\}$$

We name $A_\epsilon(j, \boldsymbol{k})$ the set of predecessors of $(j, \boldsymbol{k})$, i.e.

$$A_\epsilon(j, \boldsymbol{k}) = \{(j', \boldsymbol{k}') : j' < j \text{ and } B_{j'\boldsymbol{k}'} \cap b_\epsilon(j, \boldsymbol{k}) \neq \emptyset\}$$

The cardinal of $A_\epsilon(j, \boldsymbol{k})$ is finite. Moreover, since the $(\nu_{j'\boldsymbol{k}'})_{(j',\boldsymbol{k}') \in A_\epsilon(j,\boldsymbol{k})}$ are integer multiples of 2^{-j}, the balls $b_\epsilon(j', \boldsymbol{k}')$ for $(j', \boldsymbol{k}') \in A_\epsilon(j, \boldsymbol{k})$ are not overlapping as soon as $\epsilon < 2^{-j-1}$.

Since the probability of the event $\{\boldsymbol{x}_n \notin b_\epsilon(j, \boldsymbol{k})\}$ is strictly less than 1, we have for all $(j', \boldsymbol{k}')$

$$\exists n \text{ such that } \boldsymbol{x}_n \in b_\epsilon(j', \boldsymbol{k}') \qquad \text{almost surely}$$

and since $A_\epsilon(j, \boldsymbol{k})$ is finite,

$$\forall (j', \boldsymbol{k}') \in A_\epsilon(j, \boldsymbol{k}),\ \exists n_{j'\boldsymbol{k}'} \text{ such that } \boldsymbol{x}_{n_{j'\boldsymbol{k}'}} \in b_\epsilon(j, \boldsymbol{k}) \qquad \text{almost surely.}$$

We then set

$$n = \max\{n_{j,\boldsymbol{k}}\}$$

We can then prove that $A_\epsilon(j, \boldsymbol{k}) \subset I_n$ and that

$$\forall (j', \boldsymbol{k}') \in A_\epsilon(j, \boldsymbol{k}),\ |i_n^{-1}(j, \boldsymbol{k}) - \nu_{j\boldsymbol{k}}| < \epsilon \tag{8.3}$$

for example recursively on j, taking advantage of the fact that the sequence $(I_m)_{m \in \mathbb{N}}$ is increasing and that if $(j, \boldsymbol{k}) \in I_{m_0}$, then the sequence

$$m \mapsto |i_m^{-1}(j, \boldsymbol{k}) - \nu_{j,\boldsymbol{k}}| \qquad m \geq m_0$$

is nonincreasing.

From (8.3), we then get that for all $(j, \boldsymbol{k})$ and all sufficiently small ϵ (smaller than 2^{-j-1}), we have

$$\exists n,\ m \geq n \Rightarrow (j, \boldsymbol{k}) \in I_m \text{and } |i_m^{-1}(j, \boldsymbol{k}) - \nu_{j,\boldsymbol{k}}| < \epsilon \qquad \text{a.s.}$$

By using this relationship for the denumerable set of $\epsilon_k = 2^{-j-1-k}$, we obtain by addition of almost sure statements the following almost sure statement:

$$\exists n,\ m \geq n \Rightarrow (j, \boldsymbol{k}) \in I_m \text{and } i_m^{-1}(j, \boldsymbol{k}) \to \nu_{j\boldsymbol{k}} \quad \text{when } m \to +\infty$$

□

Why is this interesting? If we choose a subsystem of wavelets $(j, \boldsymbol{k})$ for $j < j_0$ (in a finite domain D), we then know that almost surely the measures allocated to each coefficient $(j, \boldsymbol{k})$ converge to the center of the corresponding wavelet. In the limit case, the considered subsystem corresponds to uniform sampling. The matrix of the subsystem is then simply the matrix of the wavelet transform for these wavelets. This matrix is triangular, invertible and well conditioned, because both its norm and the norm of its inverse are bounded (in ℓ^∞–operator norm) by some multiple of j, i.e. the log of the total number of measures.

8.4 Allocation in a Schauder basis

We will see that in the simplest possible setting (dimension 1, piecewise linear wavelets), the allocation algorithm can be used to implement an adaptive interpolation scheme. This simple example will already show us the main difficulties we will have to face in more complex cases. To begin, we recall some results on diagonally dominant matrices we will heavily use in the following sections.

8.4.1 Conditioning of diagonally dominant matrices

A matrix A of coefficients $(a_{ij})_{1 \leq i,j \leq n}$ is said to be a diagonally dominant matrix if

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}| \qquad \text{for all } i = 1 \dots n$$

According to a theorem of Hadamard, the matrix A is then invertible. An elementary proof can be given by contradiction. Assume there exists a non zero vector v such that $Av = 0$. Let v_i be a coordinate of v of maximum modulus. $Av = 0$ then implies

$$\begin{aligned} |a_{ii} v_i| &= \left| \sum_{j \neq i} a_{ij} v_j \right| \\ \text{i.e.} \quad |a_{ii} v_i| &\leq \sum_{j \neq i} |a_{ij} v_j| \\ &\leq |v_i| \sum_{j \neq i} |a_{ij}| \\ &< |a_{ii} v_i| \end{aligned}$$

and thus a contradiction.

If we want to estimate the norm of the inverse of such a diagonally dominant matrix, we have to make a slightly stronger assumption:

Definition 8.10

A matrix A of coefficients $(a_{ij})_{i,j \in I}$ (possibly of infinite size) is said to be a diagonally dominant matrix with margin η if

$$|a_{ii}| \geq \eta + \sum_{j \neq i} |a_{ij}| \qquad \textit{for all } i \in I$$

In the infinite case, we also suppose that this operator A is a continuous mapping from ℓ^∞ to ℓ^∞, which is equivalent to assuming that the sequence $(a_{ii})_{i\in I}$ is bounded.

In this case, the following theorem gives an estimate of the norm of the inverse matrix A^{-1}.

Theorem 8.4

Let A be a bounded diagonally dominant matrix with margin η. We define $a^+ = \sup_i |a_{ii}|$ and $a^- = \inf_i |a_{ii}|$. The matrix A is then invertible, and the norm of its inverse A^{-1} is bounded by

$$\|A^{-1}\| \leq \frac{a^+}{\eta a^-}$$

Proof. Let D be the diagonal matrix of coefficients $d_{ii} = a_{ii}$. D is clearly invertible and continuous. We define $B = D^{-1}A$. The matrix B of coefficients b_{ij} fulfills then

$$\begin{aligned} b_{ii} &= 1 \\ \sum_{j\neq i} |b_{ij}| &\leq 1 - \frac{\eta}{|a_{ii}|} \\ &\leq 1 - \frac{\eta}{a^+} \end{aligned}$$

We notice that necessarily, $a^+ \geq a^- \geq \eta$. We also remark that the operator norm of $I - B$ is less than or equal to $1 - \eta/a^+$ and thus strictly less than 1. B is then invertible, and its inverse is the infinite sum of the following absolutely converging series:

$$B^{-1} = \sum_{k=1}^{+\infty} (I - B)^k$$

and moreover

$$\begin{aligned} \|B^{-1}\| &\leq \sum_{k=1}^{+\infty} \|(I - B)^k\| \\ &\leq \sum_{k=1}^{+\infty} \left(1 - \frac{\eta}{a^+}\right)^k \\ &= \frac{a^+}{\eta} \end{aligned}$$

As a consequence, $A = DB$ is also invertible, and the norm of its inverse fulfills

$$\begin{aligned} \|A^{-1}\| &\leq \|D^{-1}\|.\|B^{-1}\| \\ &\leq \frac{a^+}{\eta a^-} \end{aligned}$$

□

After these preliminaries, we can tackle our main problem.

8.4.2 The Schauder basis

The basis we use here is a Schauder basis of functions defined over the interval $[0,1]$.

We denote by ϕ a piecewise linear function defined as

$$\phi(x) = \begin{cases} 0 & \text{if } x < -1 \text{ or } x > 1 \\ x+1 & \text{if } x \in [-1,0] \\ 1-x & \text{if } x \in [0,1] \end{cases}$$

and we then define the basis functions:

$$\phi_{jk}(x) = \phi(2^j x - k) \qquad \text{for } x \in [0,1]$$

The Schauder basis is the family of functions

$$\phi_{jk} \qquad \text{for } \begin{cases} j = 0 \text{ and } k \in \{0,1\} \\ j > 0 \text{ and } k \in (\mathbb{Z} - 2\mathbb{Z}) \cap \{0 \dots 2^j\} \end{cases}$$

The centers of the wavelets are $\nu_{jk} = 2^{-j}k$. The attraction basins are

$$B_{jk} = \begin{cases} \left[k - \frac{1}{2}, k + \frac{1}{2}\right] \cap [0,1] & \text{if } j = 0 \\ \left[2^{-j}(k-1), 2^{-j}(k+1)\right] & \text{if } j > 0 \end{cases}$$

When we are given a set of measures (x_n, y_n), with distinct measure points x_n, and we choose by allocation a subset of the Schauder basis, we will see that under some conditions, we will obtain an invertible system of interpolation constraints.

Let $(\phi_{jk})_{k \in \Lambda_j}$ be the subfamily of wavelets chosen by allocation. We first perform a transformation on this wavelet basis we call relocation.

8.4.3 Relocation

The relocation of a subfamily of an interpolating wavelet basis has a very general definition. It is a triangular transformation that maps a family $(\phi_{jk})_{k \in \Lambda_j}$ to a family $(\varphi_{jk})_{k \in \Lambda_j}$ in the following way:

Definition 8.11 (Relocated wavelet subfamily)

Let $(\phi_{j\boldsymbol{k}})_{\boldsymbol{k} \in \Lambda_j}$ be a subfamily of an interpolating wavelet basis. The relocated subfamily is the family of functions $(\varphi_{j\boldsymbol{k}})_{\boldsymbol{k} \in \Lambda_j}$ defined as

$$\varphi_{j\boldsymbol{k}} = \phi_{j\boldsymbol{k}} \qquad \text{if } \Lambda_{j'} = \emptyset \text{ for } j' > j \tag{8.4a}$$

Else, we define recursively the sequence of functions $(\phi_{j\boldsymbol{k};j'})_{j' \geq j}$ by

$$\begin{aligned} \phi_{j\boldsymbol{k};j} &= \phi_{j\boldsymbol{k}} \\ \phi_{j\boldsymbol{k};j'} &= \phi_{j\boldsymbol{k};j'-1} - \sum_{\boldsymbol{k}' \in \Lambda_{j'}} \phi_{j\boldsymbol{k};j'-1}(\nu_{j'\boldsymbol{k}'})\phi_{j'\boldsymbol{k}'} \qquad \text{for } j' > j \end{aligned}$$

Since $\Lambda_{j'}$ is empty for all j' larger than a given threshold, the sequence $(\phi_{j\boldsymbol{k};j'})_{j'\geq j}$ is constant for j' large enough. We then set

$$\varphi_{j\boldsymbol{k}} = \phi_{j\boldsymbol{k};\infty} \tag{8.4b}$$

Note that when definition (8.4a) applies, both definitions are equivalent, because then the sequence $(\phi_{j\boldsymbol{k};j'})_{j'\geq j}$ is constant from the beginning.

We can also note that if the chosen subfamily contains all wavelets up to a given scale, which means in the monovariate case

$$\begin{aligned} \Lambda_0 &= \mathbb{Z}\Lambda_j && = \mathbb{Z} - 2\mathbb{Z} && \text{for } 1 \leq j \leq j_0 \\ \Lambda_j &= \emptyset && && \text{for } j > j_0, \end{aligned}$$

then this basis transformation is exactly an inverse wavelet transform.

Else, in the most general case, we have the following property:

Proposition 8.1

The collocation matrix $(\varphi_{j\boldsymbol{k}}(\nu_{j'\boldsymbol{k}'}))_{\boldsymbol{k}\in\Lambda_j,k'\in\Lambda_{j'}}$ is the identity matrix.

Proof. We already know that $\phi_{j\boldsymbol{k}}(\nu_{j'\boldsymbol{k}'}) = \delta_{jj'}\delta_{\boldsymbol{k}\boldsymbol{k}'}$ if $j' \leq j$. We can that prove by a simple recursive proof that

$$\varphi_{j\boldsymbol{k};j''}(\nu_{j'\boldsymbol{k}'}) = \delta_{jj'}\delta_{\boldsymbol{k}\boldsymbol{k}'} \qquad \text{for all } j' \leq j'' \text{ and } \boldsymbol{k}' \in \Lambda'_j$$

□

Example

Suppose we have 5 measure points

$$\begin{aligned} x_1 &= 0,1 & x_2 &= 0,23 \\ x_3 &= 0,35 & x_4 &= 0,62 \\ x_5 &= 0,8 \end{aligned}$$

The allocation process will select 5 wavelets with the following mapping:

$$\begin{aligned} x_0 &\leftrightarrow \phi_{00} & x_4 &\leftrightarrow \phi_{01} \\ x_3 &\leftrightarrow \phi_{11} & x_1 &\leftrightarrow \phi_{21} \\ x_2 &\leftrightarrow \phi_{33} \end{aligned}$$

The relocated wavelets are represented in Fig. 8.6. We can show that if the measures placed in the coefficient tree fulfill an exclusion criterion we detail thereafter, the linear system of interpolation constraints is invertible.

The exclusion criterion we use is the following:

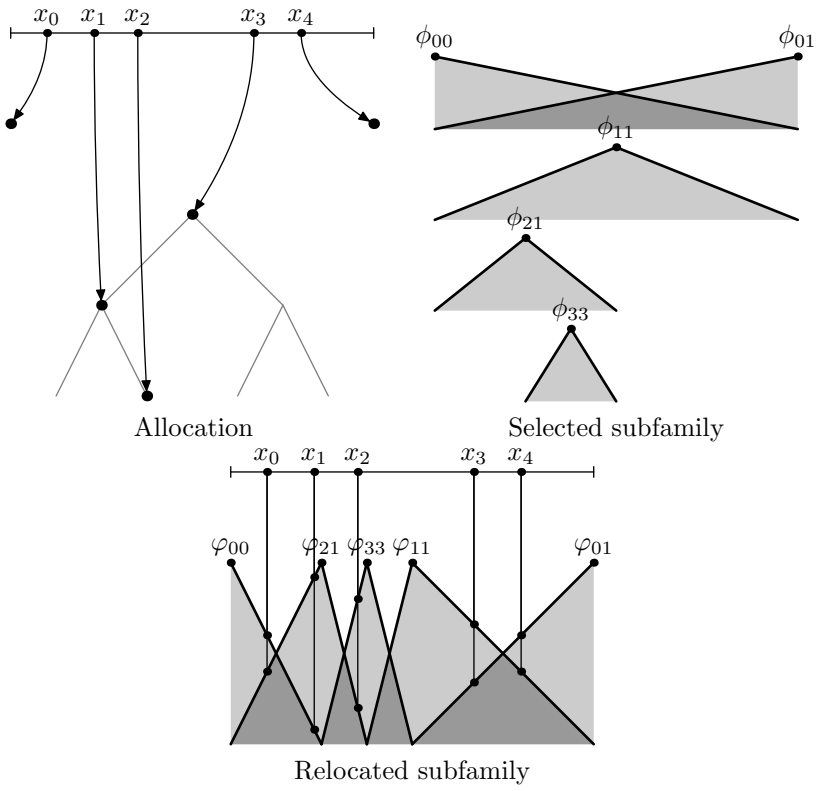


Figure 8.6: Allocation example in a dyadic interpolating wavelet basis. Relocation example.

Definition 8.12

Let u and v be two positive reals. An allocation $i : \mathcal{X} \to \mathcal{B}$ fulfills the exclusion criterion of parameters (u, v) if for all $(j, k) = i(x)$, we have the following property:

$$(j', k') = i(x') \text{and } j' < j \Rightarrow x' \notin [\nu_{jk} - 2^{-j}u; \nu_{jk} + 2^{-j}u] \qquad (\text{Ex}_1)$$

$$x \in]\nu_{jk} - 2^{-j}v; \nu_{jk} + 2^{-j}v[\qquad (\text{Ex}_2)$$

We call the interval $[\nu_{jk} - 2^{-j}u; \nu_{jk} + 2^{-j}u]$ the exclusion interval of the wavelet ϕ_{jk} of width u.

Remarks

- *We can first remark that it is easy to check whether this criterion is fulfilled, because it suffices to do this verification for only two measures x' that correspond to two wavelets (j_1, k_1) and (j_2, k_2) of resolutions j_1 and j_2 such that*

$$2^{-j_1}k_1 = 2^{-j}(k-1) \qquad 2^{-j_2}k_2 = 2^{-j}(k+1)$$

  where of course $k_1 \in K_{j_1}$ and $k_2 \in K_{j_2}$.

- *We can show that there exists a largest subtree of coefficients in the tree $I = i(\mathcal{X})$ for which the exclusion criterion holds. When the fill distance*

$$\max_{x \in [0,1]} \min_{x' \in \mathcal{X}} |x - x'|$$

  goes to 0, this subtree fulfilling the exclusion criterion converges to the whole wavelet tree, i.e.

$$\inf\{j : K_j \neq \Lambda_j\} \to +\infty$$

We now prove that an allocation that fulfills the exclusion criterion with parameters $(1/2, 1/2)$ provides us with a linear system of interpolation constraints that is invertible. If moreover u is larger than v, we can bound the norm of the system matrix inverse.

Theorem 8.5

If an allocation of a measure set (x_i, y_i) with $x_i \in [0, 1]$ into a Schauder basis fulfills the exclusion criterion with parameters $(1/2, 1/2)$, then the system of interpolation constraints is invertible. If it fulfills the exclusion criterion of parameters u, v where $u = 1/2 + \eta$ and $v = 1/2 - \eta$, then the system matrix inverse has an operator ℓ^∞ norm bounded by some Mj_{max}/η.

Proof. The proof can be split in the following way. We rearrange the selected wavelets $(\phi_{jk})_{(j,k)\in I}$ so that their centers ν_{jk} are rearranged in increasing order, into

$$(\phi_{j_t k_t})_{t=1\ldots N}$$

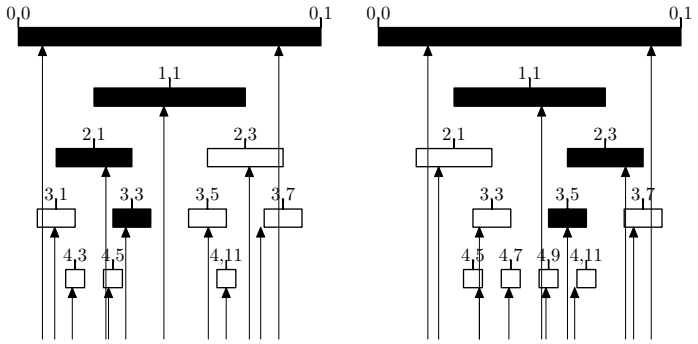


Figure 8.7: The two figures above represent allocation examples into the Schauder basis. In either case, the selected wavelets are marked with a black or white rectangle. Each measure point is indicated with an arrow whose abscissa is the location of the measure point, and whose head indicates which wavelets it is allocated to. The width of the rectangle indicates the exclusion interval of the wavelet. Wavelets that make up the largest tree fulfilling the exclusion criterion are marked with a black rectangle. We see for example in the left figure (a) that wavelet ϕ_{31} does not fulfill the exclusion criterion because the measure allocated to the wavelet ϕ_{00} is in its exclusion interval (see (Ex_1)). For the same reason, the wavelets ϕ_{23} and its descendants are also marked with a white rectangle. In figure (b), we have an example of the second reason why a wavelet may not fulfill the exclusion criterion: the measure allocated to the wavelet $(4, 11)$ is not in its exclusion rectangle (see (Ex_2)).

where $N = \operatorname{card} \mathcal{X}$, so that finally

$$\nu_{j_1 k_1} < \cdots < \nu_{j_N k_N}$$

To simplify this proof, we suppose that the wavelets of resolution $j = 0$ are already in I[1], so that

$$\nu_{j_1 k_1} = 0 \qquad\qquad \nu_{j_N k_N} = 1$$

and for practical notation reasons, we also assume we can write:

$$\nu_{j_0 k_0} = 0 \qquad\qquad \nu_{j_{N+1} k_{N+1}} = 1$$

We show that the corresponding relocated wavelets φ_{jk} are piecewise linear (with only two "pieces"). A wavelet $\varphi_{j_t k_t}$ is thus defined as

$$\varphi_{j_t k_t}(x) = \begin{cases} 0 & \text{for } t \in [0, \nu_{j_{t-1} k_{t-1}}[\\ 1 & \text{if } t = \nu_{j_t k_t} \\ 0 & \text{for } t \in]\nu_{j_{t+1} k_{t+1}}, 1] \end{cases}$$

for $t = 1 \ldots N$ and is linear on both intervals $[\nu_{j_{t-1} k_{t-1}}, \nu_{j_t k_t}]$ and $[\nu_{j_t k_t}, \nu_{j_{t+1} k_{t+1}}]$. An example of such a relocalized family is shown in Fig. 8.6–c.

Lemma 8.1

We denote by x_{jk} the measure allocated to a wavelet ϕ_{jk}, i.e. such that $i(x_{jk}) = (j, k)$. If the allocation fulfills the exclusion criterion, we can then show that each measure $x_{j_t k_t}$ is in the interval

$$\frac{\nu_{j_{t-1} k_{t-1}} + \nu_{j_t k_t}}{2} < x_{j_t k_t} < \frac{\nu_{j_t k_t} + \nu_{j_{t+1} k_{t+1}}}{2} \tag{8.5}$$

Proof. The proof can be done recursively. The recursion is started with the case when $I = \{(0, 0), (0, 1)\}$, i.e. when the only selected wavelets are those of resolution $j = 0$. In this simple case, we can easily show that inequality (Ex_2) for each wavelet implies (8.5).

Let us now consider a subtree I to which we have allocated a set of measures $x \in \mathcal{X}$, and assume that this subtree fulfills the exclusion criterion. We assume that the wavelets are arranged by increasing center position into $\phi_{j_t k_t}$.

We choose a wavelet $\phi_{j_t k_t}$ of maximum resolution j_t in I. The recursive statement applies to the subfamily of indices $I - \{(j_t, k_t)\}$. For $\tau \notin \{t - 1, t, t + 1\}$, we then have

$$\frac{\nu_{j_{\tau-1} k_{\tau-1}} + \nu_{j_\tau k_\tau}}{2} < x_{j_\tau k_\tau} < \frac{\nu_{j_\tau k_\tau} + \nu_{j_{\tau+1} k_{\tau+1}}}{2}$$

and also

$$\frac{\nu_{j_{t-2} k_{t-2}} + \nu_{j_{t-1} k_{t-1}}}{2} < x_{j_{t-1} k_{t-1}} < \frac{\nu_{j_{t-1} k_{t-1}} + \nu_{j_{t+1} k_{t+1}}}{2}$$
$$\frac{\nu_{j_{t-1} k_{t-1}} + \nu_{j_{t+1} k_{t+1}}}{2} < x_{j_{t+1} k_{t+1}} < \frac{\nu_{j_{t+1} k_{t+1}} + \nu_{j_{t+2} k_{t+2}}}{2}$$

[1] although the proof can be written without assuming this.

We then have to complete this chain of inequalities by proving that

$$x_{j_{t-1}k_{t-1}} < \frac{\nu_{j_{t-1}k_{t-1}} + \nu_{j_tk_t}}{2} \tag{8.6}$$

$$\frac{\nu_{j_{t-1}k_{t-1}} + \nu_{j_tk_t}}{2} < x_{j_tk_t} < \frac{\nu_{j_tk_t} + \nu_{j_{t+1}k_{t+1}}}{2} \tag{8.7}$$

$$\frac{\nu_{j_tk_t} + \nu_{j_{t+1}k_{t+1}}}{2} < x_{j_{t+1}k_{t+1}} \tag{8.8}$$

Since the family of selected coefficients I makes up a tree, we can check that

$$2^{-j_t}(k_t+1) = 2^{-j_{t+1}}k_{t+1}$$
$$2^{-j_t}(k_t-1) = 2^{-j_{t-1}}k_{t-1}$$

The exclusion criterion applied for t then implies by (Ex_1) for $j = j_t$ and $j' = j_{t-1}$ the inequality (8.6). Indeed, we know by recursion that

$$x_{j_{t-1}k_{t-1}} < \frac{\nu_{j_{t-1}k_{t-1}} + \nu_{j_{t+1}k_{t+1}}}{2}$$

and thanks to (Ex_1)

$$x_{j_{t-1}k_{t-1}} \notin \left[\nu_{j_tk_t} - 2^{-j_t-1}, \nu_{j_tk_t} + 2^{-j_t-1}\right]$$

which implies (8.6). We obtain (8.8) in a similar way. The double inequality (8.7) is obtained by application of (Ex_2) for $j = j_t$. □

As a consequence, for each measure $x_{j_tk_t}$, $t = 1 \dots N$, there exist at most two relocated wavelets that have a non zero value at that location. The first one is always $\varphi_{j_tk_t}$ which has a value strictly larger than $1/2$, while the second one is either $\varphi_{j_{t-1}k_{t-1}}$, or $\varphi_{j_{t+1}k_{t+1}}$ which has on $x_{j_tk_t}$ a value smaller than $1/2$. As a consequence, the system matrix is a diagonally dominant matrix, and is thus invertible.

If we now assume that the allocation fulfills the exclusion criterion of parameters (u, v) where $u = 1/2 + \eta$, $v = 1/2 - \eta$ and $\eta > 0$ then we can prove in a similar way that each measure point is located in the interval defined by the inequalities

$$\left(\frac{1}{2} - \eta\right)\left(\nu_{j_{t-1}k_{t-1}} - \nu_{j_tk_t}\right) < x_{j_tk_t} - \nu_{j_tk_t} < \left(\frac{1}{2} - \eta\right)\left(\nu_{j_{t+1}k_{t+1}} - \nu_{j_tk_t}\right)$$

In this case, the matrix A' of the interpolation constraints in the relocated wavelet subfamily is diagonally dominant with a strictly positive margin. Indeed,

$$\varphi_{j_tk_t}(x_{j_tk_t}) > \frac{1}{2} + \eta$$
$$\varphi_{j_{t-1}k_{t-1}}(x_{j_tk_t}) < \frac{1}{2} - \eta$$
$$\varphi_{j_{t+1}k_{t+1}}(x_{j_tk_t}) < \frac{1}{2} - \eta$$

which leads to:

$$|\varphi_{j_t k_t}(x_{j_t k_t})| \geq \sum_{t' \neq t} |\varphi_{j'_t k'_t}(x_{j_t k_t})| + 2\eta$$

because in this case also, either $\varphi_{j_{t-1} k_{t-1}}$ or $\varphi_{j_{t+1} k_{t+1}}$ vanishes on $x_{j_t k_t}$. The inverse matrix A'^{-1} has thus an operator norm (from ℓ^∞ to ℓ^∞) that is less than

$$\|A'^{-1}\| \leq \frac{1}{2\eta}$$

Since the norm of the basis transformation matrix corresponding to the change $(\phi_{jk}) \leftrightarrow (\varphi_{jk})$ is less that some $Mj_{\max}$, we finally get that the inverse matrix of the system of interpolation constraints A in the original wavelet subfamily is bounded by

$$\|A^{-1}\| \leq \frac{Mj}{2\eta}$$

□

The exclusion criterion most notably implies that the considered measure points be sufficiently far apart. More precisely, one can verify that the distance between two measure points x_{jk} and $x_{j'k'}$ is necessarily larger than $2\eta 2^{-\max(j,j')}$, that is lower bounded by some multiple of finest resolution step between 2^{-j} and $2^{-j'}$.

Intuitively, it seems fairly natural that if two measures are close one to another, at least one of them has to be allocated to a wavelet of scale 2^{-j} comparable to the distance between these measures in order for the system of constraints to be stable. Else, we would try to explain small scale variations of the functions with basis functions that only have coarse scale variations, and make up an unstable linear system.

We will see that such a hypothesis on the distance between measures is not sufficient to guarantee that the linear system is stable.

Let us for example consider the case of three measures $x_1 = 0$, $x_2 = 1/2+\epsilon$ and $x_3 = 1/4$. The wavelets associated to these measures by the allocation algorithm are ϕ_{00}, ϕ_{01} and ϕ_{11}. The functions that can be represented with such basis functions are piecewise linear functions on the intervals $[0, 1/2]$ and $[1/2, 1]$. As ϵ goes to 0, the system of linear constraints is getting degenerate, although the relative distance between any two measures stays lower bounded. Indeed, by changing y_3 by an amount of h, the coefficient of ϕ_{01} varies by an amount of $2h/\epsilon$, and the norm of system inverse matrix can be arbitrarily large.

8.5 Extension to higher dimensions

In this section, we extend the above results to higher dimension and to higher wavelet smoothness and approximation orders. We concentrate on how to ensure that a given choice of wavelets provides a stable system of interpolation constraints. We give sufficient conditions for this in 8.5.2. We then prove the resulting stability theorem in Sec. 8.5.1. This theorem contains two main parts:

- The L^∞ norm of the interpolated function $f_\mathcal{X}$ obtained from the samples $(\boldsymbol{x}, f(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$ is bounded by some constant times $\sup_{\boldsymbol{x}\in\mathcal{X}} |f(\boldsymbol{x})|$. As a trivial consequence, we have a bound

$$\|f_\mathcal{X}\|_\infty \leq M \times \|f\|_\infty$$

- The operator norm of the inverse matrix of the constraint system is very low. If the system matrix is written A, and if $j_{\max}$ is the largest resolution index in the selected wavelet subfamily, then

$$\|A^{-1}\|_{\infty,\infty} \leq M j_{\max}$$

  As a consequence, if $q(\mathcal{X})$ is the separation distance of the set $\mathcal{X}$, i.e.

$$q(\mathcal{X}) = \min_{\substack{\boldsymbol{x},\boldsymbol{x}'\in\mathcal{X} \\ \boldsymbol{x}\neq\boldsymbol{x}'}} \|\boldsymbol{x} - \boldsymbol{x}'\|$$

  then the norm of A^{-1} is bounded by some

$$\|A^{-1}\|_{\infty,\infty} \leq M_1 + M_2 |\log q(\mathcal{X})|$$

Some parts of the tools we have presented above can be reused to generalize these results to interpolating wavelets of any order and domains of any dimension, while some others will have to be replaced.

The exclusion criterion cannot be reused as such in higher dimensions. Indeed, we have implicitly used the fact that building a dyadic tree consists in doing a hierarchical partitioning of the segment $[0, 1]$, which is not the case any more in higher dimensions.

The exclusion criterion will be replaced by another selection criterion we call a criterion of "good relative positioning".

Definition 8.13 (Good relative positioning (GRP) criterion)

Let i be an allocation of a finite set $\mathcal{X}$ of measure points into an interpolating wavelet basis. We say that a measure point $\boldsymbol{x}$ allocated to a wavelet $\phi_{j\boldsymbol{k}}$ fulfills a criterion of good relative positioning of parameters $p \in \mathbb{N}$ and $\rho \in \mathbb{R}$ if for all $(j', \boldsymbol{k}')$

$$\left.\begin{array}{c} \|\nu_{j'\boldsymbol{k}'} - \nu_{j\boldsymbol{k}}\|_\infty \leq 6p2^{-j} \\ j' < j \end{array}\right\} \Rightarrow \left\{\begin{array}{c} \exists \boldsymbol{x}' \in \mathcal{X} : i(\boldsymbol{x}') = (j', \boldsymbol{k}') \\ \|\boldsymbol{x}' - \nu_{j'\boldsymbol{k}'}\|_\infty \leq \rho 2^{-j} \end{array}\right. \tag{8.9}$$

We then say that an allocation fulfills the good relative positioning criterion whenever all the allocated measures fulfill the above criterion.

Remarks

- *This criterion implies that a measure allocated to a wavelet of indices $(j, \boldsymbol{k})$ can only be used if*

  1. *the wavelets whose centers are close to $2^{-j}\boldsymbol{k}$ and are of lower resolution also are in $\{\phi_{j'\boldsymbol{k}'} : (j', \boldsymbol{k}') \in I\}$.*

2. *and for these wavelets, the distance between the allocated measure point and the wavelet center is smaller than some constant times* 2^{-j}.

- *We can first notice that checking whether a given wavelet of indices* $(j, \boldsymbol{k})$ *fulfills this criterion only requires to look at wavelets of scale coarser than* j. *If we are given an allocation, we can thus extract a largest subtree of coefficients fulfilling the good relative positioning criterion.*

- *The computational cost of extracting such a subtree is relatively small, since for each wavelet, we only have to consider at most a fixed number of* $(2p+2)^d$ *other wavelets. The total cost of extraction of the largest subtree is thus bounded by some multiple of the number of measures* $\mathcal{O}(N)$.

The criterion of good relative positioning will allow us to select a subtree for which we can prove that the resulting system of interpolation constraints is invertible and well conditioned.

8.5.1 Conditions of stability

In this section, we prove that if the allocation we consider fulfills the criterion for good relative positioning with parameters p and ρ we define hereafter, the system of interpolation constraints is stable. To choose a parameter ρ, we give the following definition.

Let P be the set $\{-p \dots p+1\}^d$ and $P' = P \cap (2\mathbb{Z}^d)$. For all $\boldsymbol{k} \in P$ and $S \subset P'$, we define

$$\phi^S_{0\boldsymbol{k}} = \phi_{0\boldsymbol{k}} - \sum_{\boldsymbol{k}' \in S} \phi_{0\boldsymbol{k}}(2^{-1}\boldsymbol{k}')\phi_{1\boldsymbol{k}'}$$

This consists in taking the wavelets of resolution 0 or 1 whose support overlaps the hypercube $[-1/4; 3/8]^d$. We suppose that only a subset of the wavelets of scale 1 are selected (say by allocation), and the above notation of ϕ^S refers to relocated wavelets. The formula defining these wavelets is relatively simple since their computation only involves corrections at a single scale.

We can then define the dominance radius ρ.

Definition 8.14 (Dominance radius of margin η.)

Let ϕ *be an interpolation wavelet of compact support (included in some* $[-p, p]$*). Let* $(\phi_{j\boldsymbol{k}})_{\boldsymbol{k} \in K_j}$ *be a wavelet basis defined over* $\mathbb{R}^d$ *by the following construction:*

$$\begin{aligned} K_0 &= \mathbb{Z}^d \\ K_j &= \mathbb{Z}^d - 2\mathbb{Z}^d \qquad\qquad \text{for } j > 0 \end{aligned}$$

and by

$$\phi_{j\boldsymbol{k}}(\boldsymbol{x}) = \phi(2^j x_1 - k_1) \times \cdots \times \phi(2^j x_d - k_d)$$

We then define

$$D_{0,\eta} = \left\{ \boldsymbol{x} : |\phi^S_{00}(\boldsymbol{x})| \geq \eta + \sum_{\boldsymbol{k}\in P-\{0\}} |\phi^S_{0\boldsymbol{k}}(\boldsymbol{x})| + \sum_{\boldsymbol{k}\in P'} |\phi_{1\boldsymbol{k}}(\boldsymbol{x})| \right\} \tag{8.10}$$

Then, for all $\boldsymbol{\epsilon} \in \{0,1\}^d$, $\boldsymbol{\epsilon} \neq 0$, *we define*

$$D_{\boldsymbol{\epsilon},\eta} = \left\{ \boldsymbol{x} : |\phi_{1\boldsymbol{\epsilon}}(\boldsymbol{x})| \geq \eta + \sum_{\boldsymbol{k}\in P} |\phi^S_{0\boldsymbol{k}}(\boldsymbol{x})| + \sum_{\boldsymbol{k}\in P'-\{\boldsymbol{\epsilon}\}} |\phi_{1\boldsymbol{k}}(\boldsymbol{x})| \right\} \tag{8.11}$$

For η *less than* $1/2$, *we know that* $0 \in D_{0,\eta}$ *and* $\boldsymbol{\epsilon}/2 \in D_{\boldsymbol{\epsilon},\eta}$. *Indeed the left hand side of the inequality that defines* $D_{0,\eta}$ *or* $D_{\boldsymbol{\epsilon},\eta}$ *is in this case* 1 *and the right hand side* η. *If then* $\eta < 1/2$, *each* $D_{\boldsymbol{\epsilon},\eta}$ *contains a vicinity of* $\boldsymbol{\epsilon}/2$.

Next, we set

$$\rho_{\boldsymbol{\epsilon},\eta} = \max\{\rho : |\boldsymbol{x} - \boldsymbol{\epsilon}/2| \leq \rho \Rightarrow \boldsymbol{x} \in D_{\boldsymbol{\epsilon},\eta}\} \tag{8.12}$$

The dominance radius of margin η *is then defined as*

$$\rho_\eta = \inf_{\boldsymbol{\epsilon}\in\{0,1\}^d} \rho_{\boldsymbol{\epsilon},\eta} \tag{8.13}$$

We can verify that if $\eta < 1/2$, this radius is strictly positive. In the case of the Schauder basis above, this radius is $(1-\eta)/4$.

8.5.2 Stability in case of good relative positioning

We can now state a first stability result.

Theorem 8.6

Let i *be an optimal allocation of a family of measures* $\boldsymbol{x} \in \mathcal{X}$ *that fulfill the criterion of good relative positioning of radius* ρ_η. *Then, the system of interpolation constraints is invertible. The norm of the inverse matrix in the subfamily of selected wavelets is bounded by some* Mj_{max}/η *where* j_{max} *is the highest resolution involved in the subfamily* I. *The norm of the inverse matrix in the relocated wavelet basis is bounded uniformly by* M/η, *which is a constant.*

Proof. We only have to show that the system of interpolation constraints in the relocated wavelet family is a diagonally dominant matrix of margin η.

Let us consider an allocation pair of a measure point $\boldsymbol{x}$ and the corresponding wavelet of indices $(j, \boldsymbol{k})$. Let j_M be the largest index j' such that

$$\exists \boldsymbol{k}' \in \Lambda_{j'} : \|2^{j'}\boldsymbol{x} - \boldsymbol{k}'\|_\infty \leq p$$

Let $(j_M, \boldsymbol{k}_M)$ be the index of such a wavelet. By application of the good relative positioning criterion to the wavelet $j_M, \boldsymbol{k}_M$, we can state that

$$j' < j_M \text{ and } \|2^{j'}\boldsymbol{x} - \boldsymbol{k}'\|_\infty \leq p \Rightarrow (j', \boldsymbol{k}') \in I$$

We can prove that the only relocated wavelets that might take non zero values in $\boldsymbol{x}$ are the wavelets $\varphi_{j'\boldsymbol{k}'}$ such that $\|\boldsymbol{k}' - 2^{j'}\nu_{j\boldsymbol{k}}\|_\infty \leq p$, and that their resolution index cannot be larger than j_M. We can thus put wavelets of higher resolution aside. For the others, we set

$$S = \{\boldsymbol{k}' \in \Lambda_{j_M-1} : \|\boldsymbol{k}' - \nu_{j\boldsymbol{k}}\|_\infty \leq p\}$$

If $j' < j_M$, a wavelet $\phi_{j'\boldsymbol{k}'}$ to which we apply the relocation process until the resolution $j_M - 1$ (which we denoted by $\phi_{j'\boldsymbol{k}';j_M-1}$) can be written as the linear combination

$$\begin{aligned}
\phi_{j'\boldsymbol{k}';j_M-1} &= \sum_{\kappa\in\mathbb{Z}^d} c_\kappa \phi_{j_M-1,\kappa} \\
\phi_{j'\boldsymbol{k}';j_M-1}(\nu_{j''\boldsymbol{k}''}) &= 0 \qquad \text{if } j'' < j_M \text{ and } (j'',\boldsymbol{k}') \neq (j,\boldsymbol{k}) \\
\phi_{j'\boldsymbol{k}';j_M-1}(\nu_{j\boldsymbol{k}}) &= 1 \qquad \text{if } j < j_M
\end{aligned}$$

By applying the relocation process until resolution j_M, we thus obtain functions $\phi_{j'\boldsymbol{k}';j_M}$ that coincide on the hypercube $H = \{\nu_{j\boldsymbol{k}}\} + 2^{-j}[-1/2;1/2]^d$ with

$$\begin{cases}
\phi^S_{0\boldsymbol{k}'}(2^{j_M-1}.) & \text{if } j' < j_M \text{ if } \|2^{j_M-1}\nu_{j'\boldsymbol{k}'} - \nu_{j\boldsymbol{k}}\|_\infty \leq p \\
0 & \text{if } j' < j_M \text{ and } \|2^{j_M-1}\nu_{j'\boldsymbol{k}'} - \nu_{j\boldsymbol{k}}\|_\infty > p \\
\phi_{1\boldsymbol{k}'}(2^{j_M-1}.) & \text{if } j' = j_M
\end{cases}$$

The next relocation steps do not affect the values of these wavelets in the hypercube H, because the higher resolution wavelets vanish on this hypercube. We can thus say in conclusion that $\varphi_{j'\boldsymbol{k}'}$ coincides on H with the function

$$\begin{cases}
\phi^S_{0\boldsymbol{k}'}(2^{j_M-1}.) & \text{if } j' < j_M \text{ and } \|2^{j_M-1}\nu_{j'\boldsymbol{k}'} - \nu_{j\boldsymbol{k}}\|_\infty \leq p \\
0 & \text{if } j' < j_M \text{ and } \|2^{j_M-1}\nu_{j'\boldsymbol{k}'} - \nu_{j\boldsymbol{k}}\|_\infty > p \\
\phi_{1\boldsymbol{k}'}(2^{j_M-1}.) & \text{if } j' = j_M \\
0 & \text{if } j' > j_M
\end{cases}$$

The criterion of good relative positioning being fulfilled for $\phi_{j_M\boldsymbol{k}_M}$ with the radius ρ_η, we can infer that

$$|\varphi_{j\boldsymbol{k}}(\boldsymbol{x})| \geq \eta + \sum_{\substack{(j',\boldsymbol{k}')\neq(j,\boldsymbol{k}) \\ \boldsymbol{k}'\in\Lambda_{j'}}} |\varphi_{j'\boldsymbol{k}'}(\boldsymbol{x})|$$

Since we can write this for all wavelet / measure pairs, we can say that the system of interpolation constraints in the relocated wavelet subfamily is a diagonally dominant matrix of margin η.

This matrix is thus invertible, and the operator norm of its inverse is bounded by some M_1/η. Since the norm of basis change matrix between original wavelets and relocated wavelets is bounded by $M_2 j_{\max}$, we can also bound the norm of the inverse matrix in the original wavelet basis by some $Mj_{\max}/\eta$ (with $M = M_1M_2$). We have to emphasize here that the constants M_1 and M do not depend on the number of measures and wavelets involved, but only on the space dimension d and the kind of monovariate interpolating function ϕ we used to build the wavelet basis. □

We can now also bound the L^∞ norm of the resulting interpolant by some multiple of the ℓ^∞ norm of the sequence of measured values y_n. For this purpose, we first prove the following lemma:

Lemma 8.2

Let i be an allocation of a set of measure points into a wavelet basis that fulfills the criterion of good relative positioning. Let $(\varphi_{j\boldsymbol{k}})_{\boldsymbol{k}\in\Lambda_j}$ be the associated family of relocated wavelets. Then

- *the relocated wavelets $\varphi_{j\boldsymbol{k}}$ are uniformly bounded by some φ_{max} that does not depend of the chosen subfamily of wavelets;*
- *for any fixed point $\boldsymbol{x}$, the number of relocated wavelets that have a non zero value at $\boldsymbol{x}$ is bounded by an integer N_{max} that does not depend on the number of allocated measures.*

Proof. The proof relies on arguments similar to the above stability proof. To settle the first statement, we take any wavelet of the relocated wavelet subfamily, and show that it can be written as a contracted form of some $\phi^S_{0\boldsymbol{k}}$.

Let $\varphi_{j\boldsymbol{k}}$ be a relocated wavelet. Let

$$j_M = \max\{j' : \exists \boldsymbol{k}'\Lambda_{j'} \text{ such that } \|2^{j'}\nu_{j\boldsymbol{k}} - \boldsymbol{k}\|_\infty \leq 2p\}$$

By application of the good relative positioning criterion, we can show by a simple recursion that for all $j' \in \{0 \dots j_M - 1\}$, we have

$$|2^{j'}\nu_{j\boldsymbol{k}} - \boldsymbol{k}'| \leq 2p \Rightarrow (j', \boldsymbol{k}') \in I$$

In this case, the wavelet $\phi_{j\boldsymbol{k}}$ relocated up to the scale $j_M - 1$ can be written as

$$\phi_{j\boldsymbol{k};j_M-1} = \phi_{j_M-1,2^{j_M-1-j}\boldsymbol{k}}$$

Since this wavelet has its support in the hypercube $H = \nu_{j\boldsymbol{k}} + 2^{-j}[-p;p]^d$, we see that $\varphi_{j\boldsymbol{k}}$ can be written as

$$\varphi_{j\boldsymbol{k}} = \phi^{S'}_{0\boldsymbol{k}}(2^j.)$$

where

$$\phi^{S'}_{0\boldsymbol{k}}(\boldsymbol{x}) = \phi_{0\boldsymbol{k}} - \sum_{\boldsymbol{k}'\in S'} \phi_0\boldsymbol{k}(\boldsymbol{k}'/2)\phi_{1\boldsymbol{k}'}$$
$$S' = \{\boldsymbol{k}' \in \Lambda_{j_M} : \|\boldsymbol{k}' - \boldsymbol{k}\|_\infty \leq 2p\}$$

The functions $\phi^S_{0\boldsymbol{k}}$ that can be obtained this way are of finite number, and are therefore uniformly bounded by some $\varphi_{\max}$. As a result, the relocated wavelets are also uniformly bounded by the same $\varphi_{\max}$.

The second point can be proved by going over the proof of the preceding theorem. We then obtain that the number of related functions that do not vanish at a point $\boldsymbol{x}$ are at most $2 \times (2p+2)^d$. □

Theorem 8.7

With the assumptions of the preceding theorem, the L^∞ norm of the interpolant $\|f\|_\infty$ is bounded by

$$\|f\|_{L^\infty} \leq \frac{M \sup_n |y_n|}{\eta} \tag{8.14}$$

Proof. The interpolating function f can be written

$$f = \sum_{\boldsymbol{k} \in K_j} c_{j\boldsymbol{k}} \varphi_{j\boldsymbol{k}}$$

According to the preceding theorem, we can write

$$|c_{j\boldsymbol{k}}| \leq \frac{M}{\eta} \sup_n |y_n|$$

Since the relocated functions are bounded (by $\varphi_{\max}$) and because for each $\boldsymbol{x} \in D$, the number of relocated wavelets that do not vanish is also bounded (by $N_{\max}$), we get

$$\|f\|_{L^\infty} \leq \frac{M \varphi_{\max} N_{\max}}{\eta} \sup_n |y_n|$$

□

Remark

The above theorems have formally been proved for wavelets defined over the whole space $\mathbb{R}^d$, and not for the boundary corrected wavelets of $[0,1]^d$. However, these proofs can be extended to the latter case, without further technical difficulty. Notations and definitions have to be substantially modified, which makes their use more difficult and the proofs less readable. We assume them in the next sections.

8.5.3 Approximation convergence rate

The above assumptions are expressed in terms of measure allocations that fulfill a good relative positioning criterion.

In this section, we prove that this criterion is fulfilled for a tree of any depth as soon as the density of provided measure points is sufficient. In a second phase, we give some asymptotic results for the scattered interpolation scheme we have just described.

Existence of allocations fulfilling the good relative positioning criterion

Let $\mathcal{X}$ be a set of points. If the fill distance is small enough, we can prove the existence of a suballocation of any depth that fulfills the good relative positioning criterion. For this, we first give a number of definitions.

Definition 8.15 (Suballocation)

Let i be an allocation of a set of measure points $\mathcal{X}$ in a wavelet basis $\mathcal{B} = (\phi_{j\boldsymbol{k}})_{\boldsymbol{k}\in K_j}$. A pair $(\mathcal{X}', i')$ is a suballocation of i if

$$\mathcal{X}' \subset \mathcal{X}$$
$$i|_{\mathcal{X}'} = i'$$

We can easily check that if i is an optimal allocation, i' also is.

Definition 8.16

*The **depth** of an allocation or of a suballocation is the largest j such that $\Lambda_j = K_j$.*

Such a definition only makes sense (in the case of finite set of measures) if the domain D is bounded (i.e. is $[0,1]^d$ and not $\mathbb{R}^d$).

Definition 8.17 (Fill distance)

The fill distance of a set of measure points is the real $h(\mathcal{X})$ defined as

$$h(\mathcal{X}) = \max_{\boldsymbol{x}\in D} \min_{\boldsymbol{x}'\in\mathcal{X}} \|\boldsymbol{x} - \boldsymbol{x}'\|_\infty \tag{8.15}$$

We can show that if the fill distance of a set of measure points $\mathcal{X}$ is less than $\rho_\eta 2^{-j}$, then the depth of the largest optimal suballocation that fulfills the good relative positioning criterion is not less than j.

We can now state a series of theorems to estimate the convergence rate of the function approximation.

approximation of continuous functions

We can show a first theorem that applies to only continuous functions.

Theorem 8.8

Let f be a continuous function mapping $[0,1]^d$ to some normed vector space F. We denote by $f_\mathcal{X}$ the function obtained by scattered interpolation on the samples $(\boldsymbol{x}, f(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$. Then, as the fill distance $h(\mathcal{X})$ goes to 0, $f_\mathcal{X}$ converges uniformly to f.

Proof. We use as a lever a theorem of continuous function approximation in interpolating wavelet bases we recall now : if f is continuous on $[0,1]^d$, then the expansion of f in this wavelet basis truncated to the resolution range $\{0,\dots,j\}$ converges uniformly to f as $j \to +\infty$:

$$f_j = \sum_{\substack{\boldsymbol{k}'\in K_{j'} \\ j'\le j}} \langle f, \tilde{\phi}_{j'\boldsymbol{k}'}\rangle \phi_{j'\boldsymbol{k}'}$$
$$\|f - f_j\|_{L^\infty} \to 0 \qquad \text{as } j \to +\infty$$

We then apply the stability theorem to the difference $f - f_j$. Let $\epsilon > 0$. For j large enough,

$$\|f - f_j\|_{L^\infty} \leq \epsilon$$

We denote by $f_{j,\mathcal{X}}$ the interpolant obtained with the samples $(\boldsymbol{x}, f_j(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$. For a small enough fill distance $h(\mathcal{X})$, the largest suballocation fulfilling the good relative placement criterion contains all the wavelets whose resolution is less or equal to j. In this case, we have $f_{j,\mathcal{X}} = f_j$ because the solution to the interpolating linear system is unique. As a consequence,

$$\begin{aligned}
\|f_{\mathcal{X}} - f\|_{L^\infty} &\leq \|f_{\mathcal{X}} - f_j\|_{L^\infty} + \|f_j - f\|_{L^\infty} \\
&= \|f_{\mathcal{X}} - f_{j,\mathcal{X}}\|_{L^\infty} + \|f_j - f\|_{L^\infty} \\
&\leq \frac{M}{\eta} \sup_{\boldsymbol{x}\in\mathcal{X}} |f(\boldsymbol{x}) - f_j(\boldsymbol{x})| + \|f_j - f\|_{L^\infty} \\
&\leq \left(\frac{M}{\eta} + 1\right) \epsilon
\end{aligned}$$

and the L^∞ error goes to 0. □

We see that the assumptions of this theorem on the function f are weak. If the function f is not continuous, then no uniform convergence can take place.

We now tackle the case of smoother functions, for which we can provide an explicit estimate of the error.

Theorem 8.9

Let f be a function mapping $[0,1]^d$ to some normed vector space F, which is uniform α–Lipschitz on $[0,1]^d$. We then have the following interpolation error bound (by reusing the notation of the preceding theorem):

$$\|f_{\mathcal{X}} - f\|_{L^\infty} \leq M h(\mathcal{X})^\alpha \tag{8.16}$$

which holds whenever the dual interpolation wavelet has at least $\lfloor \alpha + 1 \rfloor$ vanishing moments.

Proof. We use the same approach as before. Our lever is this time a result of uniform approximation by a wavelet expansion up to a scale j of uniformly α–Lipschitz function. If f is uniformly α–Lipschitz, and if the dual wavelet has at least $\lfloor \alpha + 1 \rfloor$ vanishing moments, then the wavelet coefficients fulfill:

$$\langle f, \tilde{\phi}_{j\boldsymbol{k}} \rangle = \mathcal{O}(2^{-\alpha j})$$

and thus

$$\|f - f_j\|_{L^\infty} = \mathcal{O}(2^{-\alpha j})$$

We can write

$$\begin{aligned}
\|f - f_{\mathcal{X}}\|_\infty &\leq \|f - f_j\|_\infty + \|f_j - f_{\mathcal{X}}\|_\infty \\
&= \|f - f_j\|_\infty + \|f_{j,\mathcal{X}} - f_{\mathcal{X}}\|_\infty \\
&= \mathcal{O}(2^{-\alpha j})
\end{aligned}$$

by application of theorem 8.7 to the difference $f_j - f$, and the proof is complete. □

These results can already be compared to those of regularization briefly mentioned in Sec. 7.6.2. We can note that the error decay rate is the smaller of the Lipschitz regularity exponent and the number of vanishing moments (less 1). Whatever the situation, the norm of linear system of interpolation constraints is bounded by some Mj/η where η is the chosen diagonal dominance margin.

Conversely, regularization requires that the Lipschitz regularity exponent α of the function to estimate be known as precisely as possible. If the regularity exponent is underestimated, this impairs the error decay rate (which is the same as for wavelet interpolation). If however the regularity exponent is overestimated, then regularization fails and the system conditioning explodes, while our method converges which the best possible decay rate: that determined by the function regularity of $h(\mathcal{X})^\alpha$.

8.5.4 Non uniform approximation theorems

In this section, we show theorems stating that the error decay in our scattered interpolation process locally depends on the function smoothness. This incidentally allows us to prove convergence results for non continuous functions.

For this, we have to get back to the stability theorem 8.4. This theorem gives us a uniform bound of the coordinates of a vector x fulfilling $Ax = y$ on the basis of a uniform bound on y. This theorem states that

$$\|x\|_\infty \leq \frac{M}{\eta}\|y\|_\infty$$

if the matrix A is that of a continuous linear operator and is diagonally dominant with margin η. If we want to get more local error estimates of our interpolation process, we first have to refine the above result to get some local bound on the x coordinates depending on the y coordinates, something like:

$$|x_i| \leq \frac{M}{\eta}|y_i|$$

Unfortunately, such a simple bound cannot be expected unless the matrix A is a diagonal matrix, which is a too trivial case.

Let us first consider an extremely simple case to get more insight in what we are in order to expect. Let us suppose that y has only one component: $y = (y_i)_{i\in\mathbb{Z}}$ and $y_i = \delta_{i,0}$. Let us also suppose that the diagonal coefficients of A be all ones : $a_{ii} = 1 \forall i \in \mathbb{Z}$. The matrix $I - A$ then has an operator norm less that $1 - \eta$. Moreover, A is a band matrix of width $2p + 1$:

$$|i - j| > p \Rightarrow a_{ij} = 0$$

Then,

$$\begin{aligned} x &= A^{-1}y \\ &= \sum_{k\in\mathbb{Z}} (I - A)^k y \end{aligned}$$

The matrix $(I-A)^k$ is a band matrix of width $2kp+1$. So $(I-A)^k y$ is supported on $\{-kp,\dots,kp\}$ and is bounded by $(1-\eta)^k$. We thus get the local bound on the x coordinates:

$$\begin{aligned} |x_i| &\leq \sum_{kp\geq|i|} (1-\eta)^k \\ &\leq \frac{(1-\eta)^{|i|/p}}{\eta} \end{aligned}$$

The vector x can therefore be of infinite support, but its coefficients have an exponential decay. We will show a similar result for scattered interpolation. A local singularity will have an impact on the local estimation error that decays exponentially with the distance to the singularity.

Decay of the singularity impact

We introduce for this purpose a number of notations. We suppose we have an allocation of a set of measure points that fulfills the good relative positioning criterion with parameters p and ρ_η. The system matrix is then diagonally dominant with margin η. We define

$$j_b = \max\{j \leq 0 : K_j = \Lambda_j\} \tag{8.17}$$

which means that the allocated wavelets include all wavelets up to the level j_b.

We define the set of selected wavelet centers

$$\mathcal{N} = \{\nu_{j\boldsymbol{k}} : j \geq 0 \text{ and } \boldsymbol{k} \in \Lambda_j\} \tag{8.18}$$

We remind that the correspondence between centers and wavelets of the basis is a bijection, since

$$\left.\begin{array}{c} \nu_{j\boldsymbol{k}} = \nu_{j'\boldsymbol{k}'} \\ \boldsymbol{k} \in K_j \\ \boldsymbol{k}' \in K_{j'} \end{array}\right\} \Rightarrow \left\{\begin{array}{l} j = j' \\ \boldsymbol{k} = \boldsymbol{k}' \end{array}\right.$$

We can thus identify pairs $(j,\boldsymbol{k})$ (with $\boldsymbol{k} \in K_j$) and centers $\nu_{j\boldsymbol{k}}$. We therefore denote ϕ_ν the wavelet of center ν and $\boldsymbol{x}_\nu$ the measure point allocated to the same wavelet.

The matrix A of the system of interpolating constraints in the family of relocated wavelets thus has coefficients $a_{\nu\nu'}$ where

$$a_{\nu\nu'} = \varphi_\nu(\boldsymbol{x}_{\nu'})$$

Definition 8.18 (Adjacent centers)

One says that two wavelet centers $\nu,\nu' \in \mathcal{N}$ are adjacent if

$$a_{\nu\nu'} \neq 0 \text{ where } a_{\nu'\nu} \neq 0$$

and we denote this relationship with $\nu \leftrightarrow \nu'$.

If $\nu = \nu'$ then $\nu \leftrightarrow \nu'$, since the matrix A has no vanishing diagonal coefficient.

Definition 8.19 (Access time)

The access time between two centers is defined as the smallest t such that there exist $t+1$ adjacent centers joining them, i.e. there exists a sequence $\nu_0, \dots, \nu_t$ such that

$$\begin{aligned} &\nu_k \leftrightarrow \nu_{k+1} && \text{for } k = 0 \dots t-1 \\ &\nu_0 = \nu \\ &\nu_t = \nu' \end{aligned}$$

The access time between two centers is therefore

- zero if these centers are equal
- equal to 1 if they are different and adjacent.

Note that this definition depends on the relocated wavelet family and thus on the current allocation, and on the system matrix (and thus on the chosen wavelets).

We prove a first proposition to give a relationship between access time and center distance:

Proposition 8.2

Let an allocation be given, that fulfills the criterion of good relative positioning of parameters p and ρ_η, and let j_b be the resolution index defined in (8.17). The wavelet ϕ is supposed to have its support included in $[-p, p]$. The access time between two centers is then lower bounded by

$$t_{\nu\nu'} \geq 2^{j_b} \frac{\|\nu - \nu'\|_\infty}{p+1}$$

Proof. We give an outline of the proof. We take advantage of the fact that the wavelets of the relocated basis can be written as a wavelet of scale j_b that is further relocated (by removing components of scales higher than j_b). The relocated wavelets therefore all have their support included in some hypercube $\nu + [-p2^{-j_b}, p2^{-j_b}]^d$. Therefore, two centers ν and ν' can only be adjacent if their relative distance is less than $(p+1)2^{-j_b}$:

$$\nu \leftrightarrow \nu' \Rightarrow \|\nu - \nu'\|_\infty \leq (p+1)2^{-j_b}$$

By iterating, we obtain

$$t_{\nu\nu'} \geq 2^{j_b} \frac{\|\nu - \nu'\|_\infty}{p+1}$$

□

Let $(y_\nu)_{\nu \in \mathcal{N}}$ be a vector of support $S \subset \mathcal{N}$ and of norm $\|y\|_\infty = 1$. Let c be the vector $c = A^{-1}y$. The next proposition allows us to bound separately coefficients c_ν. For all $\nu \in \mathcal{N}$, we set

$$d(\nu, S) = \inf_{\nu' \in S} \|\nu - \nu'\|_\infty$$

Proposition 8.3

We set

$$a^+ = \sup_i a_{ii}$$
$$a^- = \inf_i a_{ii}$$

and we then have

$$|c_\nu| \leq \frac{\|y\|_\infty a^+}{\eta a^-} \exp\left(-\frac{2^{j_b} d(\nu, S)}{p+1} \ln\left(\frac{1}{1-\eta/a^+}\right)\right)$$

Proof. We define the matrix D as the diagonal matrix that has the same diagonal coefficients as A. Its coefficients thus are

$$d_{\nu\nu'} = a_{\nu\nu'}\delta_{\nu\nu'}$$

This matrix is invertible. We define

$$B = D^{-1}A$$

The matrix B now has diagonal coefficients all equal to 1 : $b_{ii} = 1$ and is diagonally dominant with margin η/a^+. We remark that in the case of the Schauder basis, the wavelets were all bounded by 1, which may be not the case here if we use a bounded domain, because some boundary wavelets have an overshoot where they take values larger that 1. As in the proof of theorem 8.4, we know that the matrix $I - B$ has a norm less than $1 - \eta/a^+$, and is thus invertible.

The same way as in the proof of theorem 8.4, we can write

$$c = \sum_{k=0}^{+\infty} (I - B)^k D^{-1} y$$

We denote by c^k the vector

$$c^k = (I - B)^k D^{-1} y$$

we can bound the ℓ^∞ norm of c^k by

$$\|c^k\|_\infty \leq \frac{1}{a^-}\left(1 - \frac{\eta}{a^+}\right)^k \|y\|_\infty$$

Moreover, a coefficient c^k_ν of this vector is non zero only if the access time between ν and S is shorter than k:

$$\begin{aligned} c^k_\nu \neq 0 &\Rightarrow \exists \nu' \in S : t_{\nu\nu'} \leq k \\ &\Rightarrow d(\nu, S) \leq 2^{-j_b}(p+1)k \end{aligned}$$

We then get the two relationships:

$$\begin{aligned} |c_\nu^k| &\leq \frac{1}{a^-}\left(1-\frac{\eta}{a^+}\right)^k \|y\|_\infty \\ c_\nu^k &= 0 \qquad\qquad\qquad\qquad \text{if } d(\nu,S) > 2^{-j_b}(p+1)k \end{aligned}$$

from which we get

$$\begin{aligned} |c_\nu| &\leq \sum_{k\geq 2^{j_b} d(\nu,S)/(p+1)} \frac{1}{a^-}\left(1-\frac{\eta}{a^+}\right)^k \|y\|_\infty \\ &\leq \frac{\|y\|_\infty a^+}{\eta a^-} \exp\left(-\frac{2^{j_b} d(\nu,S)}{p+1} \ln\left(\frac{1}{1-\eta/a^+}\right)\right) \end{aligned}$$

□

By grouping under general constants $M, \ldots$ all the parameters that only depend on the wavelets, the domain shape and the space dimension d, we can write in a shorter form

$$|c_\nu| \leq M\|y\|_\infty exp\left(-M' 2^{j_b} d(\nu,S)\right)$$

We can now transform this theorem that locally bounds expansion coefficients into a theorem that locally bounds an interpolated function.

Theorem 8.10

Let Ω be an open set of D, and f be a bounded function of support included in Ω. For any set $\mathcal{X}$ of measure points, we denote by $f_\mathcal{X}$ the function obtained with our interpolation scheme from the samples $(\boldsymbol{x}, f(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$. We define again the fill distance of $\mathcal{X}$ to be:

$$h(\mathcal{X}) = \max_{\boldsymbol{x}\in D} \min_{\boldsymbol{x}'\in\mathcal{X}} \|\boldsymbol{x}-\boldsymbol{x}'\|_\infty$$

We then have the bound

$$|f_\mathcal{X}(\boldsymbol{x})| \leq Me^{-M' d(\boldsymbol{x},\Omega)/h(\mathcal{X})}\|f\|_\infty$$

Proof. we set $j = \ln(h(\mathcal{X})/\rho_\eta)/\ln 2$. The subtree of coefficients fulfilling the good relative positioning criterion contains all wavelet coefficients up to the scale j. Let $\boldsymbol{x}$ be a point of D, then at most $N_{\max}$ relocated wavelets do not vanish at $\boldsymbol{x}$. Let

$$N = \{(j,\boldsymbol{k}) : \phi_{j\boldsymbol{k}}(\boldsymbol{x}) \neq 0\}$$

be the set of the corresponding indices. We then have

$$(j,\boldsymbol{k}) \in N \Rightarrow d(\nu_{j\boldsymbol{k}}, S) \leq d(\boldsymbol{x},S) + h(\mathcal{X})$$

and

$$d(\nu_{j\boldsymbol{k}}, S) \leq d(\nu_{j\boldsymbol{k}}, \Omega) + h(\mathcal{X})$$

and by application of proposition 8.3

$$\Rightarrow c_{j\boldsymbol{k}} \leq Me^{-M'd(\boldsymbol{x},S)/h(\mathcal{X})}\|f\|_\infty$$

We thus get

$$|f_\mathcal{X}(\boldsymbol{x})| \leq N_{\max}\varphi_{\max}Me^{-M'd(\boldsymbol{x},S)/h(\mathcal{X})}\|f\|_\infty$$

and the proof is finished. □

This last theorem allows us to give a local error bound when approximating a function of non uniform smoothness.

Theorem 8.11

Let f be a function defined on $[0,1]^d$. We assume that this function is uniformly α–Lipschitz on D less the support S of a singularity, thus $D-S$. We also assume than on a vicinity of S the function is only uniformly β–Lipschitz, with $\beta < \alpha$. Let $\mathcal{X}$ be a finite set of measure points, and let $f_\mathcal{X}$ be the interpolating function obtained with the samples $(\boldsymbol{x}, f(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$. We then have the error bound:

$$|f(\boldsymbol{x}) - f_\mathcal{X}(\boldsymbol{x})| \leq M_1 h(\mathcal{X})^\alpha + M_2 h(\mathcal{X})^\beta e^{-M_3 d(\boldsymbol{x},S)/h(\mathcal{X})}$$

We can thus see that if we are far from the singularity, the error decays like $h(\mathcal{X})^\alpha$, and on the singularity, the error decays like $h(\mathcal{X})^\beta$. When we are close to the singularity, some error diffuses around the singularity with a decaying exponential profile. The half life distance of this exponential profile is proportional to $h(\mathcal{X})$, so that this influence shrinks when $h(\mathcal{X})$ goes to 0.

Proof. The proof is based on theorem 8.10. As the fill distance $h(\mathcal{X})$ goes to 0 the largest suballocation fulfilling the good relative positioning criterion contains all wavelet coefficients up to the scale $j = \ln(h(\mathcal{X})/\rho_\eta)/\ln 2$ (which goes to $+\infty$ as the fill distance goes to 0).

We denote by f_j the expansion of f truncated to the resolutions 0 to j:

$$f_j = \sum_{\substack{j'\leq j \\ \boldsymbol{k}'\in K_{j'}}} \langle f, \tilde{\phi}_{j'\boldsymbol{k}'}\rangle \phi_{j'\boldsymbol{k}'}$$

Since the wavelets $\phi_{j\boldsymbol{k}}$ and $\tilde{\phi}_{j\boldsymbol{k}}$ have their support included in some $2^{-j}(\boldsymbol{k}+[-p,p]^d)$ and $2^{-j}(\boldsymbol{k}+[-q,q]^d)$, the assumptions on the smoothness of f imply the following error estimate:

$$|f(\boldsymbol{x}) - f_j(\boldsymbol{x})| \leq M_1 h(\mathcal{X})^\alpha + M_2 h(\mathcal{X})^\beta 1_{(d(\boldsymbol{x},\Omega)<(p+q)2^j)} \tag{8.19}$$

We now denote by $f_{j,\mathcal{X}}$ the function interpolated on the samples $(\boldsymbol{x}_n, f_j(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$ Since the fill distance is less that $\rho_\eta 2^{-j}$, we have $f_{j,\mathcal{X}} = f_j$ because the solution to the interpolation linear system is unique. We can thus write:

$$|f(\boldsymbol{x}) - f_\mathcal{X}(\boldsymbol{x})| \leq |f(\boldsymbol{x}) - f_j(\boldsymbol{x})| + |f_j(\boldsymbol{x}) - f_\mathcal{X}(\boldsymbol{x})| \tag{8.20}$$

The first term can be bounded by equation (8.19). For the second term, we decompose the error term $f_j - f$ into:

$$\begin{array}{rcccc} f_j - f & = & (f_j - f)1_{(d(\boldsymbol{x},S)\geq(p+q)2^j)} & + & (f_j - f)1_{(d(\boldsymbol{x},S)<(p+q)2^j)} \\ & = & e_1 & + & e_2 \end{array}$$

We remind that the notation $g_{\mathcal{X}}$ stands for the result of the interpolation algorithm with the samples $(\boldsymbol{x}, g(\boldsymbol{x}))_{\boldsymbol{x}\in\mathcal{X}}$. We obtain from the above equation that

$$\begin{aligned} |f_j(\boldsymbol{x}) - f_{\mathcal{X}}(\boldsymbol{x})| &= |f_{j,\mathcal{X}}(\boldsymbol{x}) - f_{\mathcal{X}}(\boldsymbol{x})| \\ &\leq |e_{1,\mathcal{X}}(\boldsymbol{x})| + |e_{2,\mathcal{X}}(\boldsymbol{x})| \end{aligned} \tag{8.21}$$

Now, since

$$|e_1(\boldsymbol{x})| \leq M_1 h(\mathcal{X})^\alpha$$

we obtain thanks to theorem 8.9

$$|e_{1,\mathcal{X}}(\boldsymbol{x})| \leq M_1' h(\mathcal{X})^\alpha \tag{8.22}$$

and since e_2 has its support included in $\{\boldsymbol{x} : d(\boldsymbol{x}, S) < (p+q)2^j\}$ and is uniformly bounded by

$$|e_2(\boldsymbol{x})| \leq M_2 h(\mathcal{X})^\beta$$

We finally get by application of theorem 8.10

$$|e_{2,\mathcal{X}}(\boldsymbol{x})| \leq M_2' h(\mathcal{X})^\beta e^{-Md(\boldsymbol{x},S)/h(\mathcal{X})} \tag{8.23}$$

From (8.21) and from the bounds (8.22) and (8.23), we obtain

$$|f_j(\boldsymbol{x}) - f_{\mathcal{X}}(\boldsymbol{x})| \leq M_1' h(\mathcal{X})^\alpha + M_2' h(\mathcal{X})^\beta e^{-Md(\boldsymbol{x},S)/h(\mathcal{X})}$$

Finally, (8.19) and (8.20) imply

$$|f(\boldsymbol{x}) - f_{\mathcal{X}}(\boldsymbol{x})| \leq M_1'' h(\mathcal{X})^\alpha + M_2'' h(\mathcal{X})^\beta e^{-Md(\boldsymbol{x},S)/h(\mathcal{X})}$$

and the proof is complete. □

As a corollary, we can show a convergence result for approximations of functions that are only piecewise continuous. Let us consider a function f that is uniformly continuous on the open sets Ω_1 and Ω_2 such that $D - (\Omega_1 \cup \Omega_2)$ is a set of measure 0. We suppose that the function is also bounded. We can then prove that the functions $f_{\mathcal{X}}$ converge to f in L_p norm as $h(\mathcal{X}) \to 0$, for any $p \geq 1$.

To simplify the proof, we assume that the function is uniformly α–Lipschitz on Ω_1 and Ω_2 for some $\alpha > 0$. In this case, we only have to apply theorem 8.11. We get:

$$|f(\boldsymbol{x}) - f_{\mathcal{X}}(\boldsymbol{x})| \leq M_1 h(\mathcal{X})^\alpha + M_2 h(\mathcal{X})^0 e^{-Md(\boldsymbol{x},S)/h(\mathcal{X})}$$

and by integration and the Minkowski triangle inequality

$$\|f - f_{\mathcal{X}}\|_{L_p} \leq M_1' h(\mathcal{X})^{\alpha} + M_2 \left(\int_D e^{-Mpd(\boldsymbol{x},S)/h(\mathcal{X})} d\boldsymbol{x} \right)^{1/p}$$

The first term in the right hand side is $M_1' h(\mathcal{X})^{\alpha}$ and goes to 0 as $h(\mathcal{X})$ goes to 0. The second term is bounded by writing:

$$\int_D e^{-Mpd(\boldsymbol{x},S)/h(\mathcal{X})} d\boldsymbol{x} \leq m\left(\{\boldsymbol{x} : d(\boldsymbol{x},S) \leq \sqrt{h(\mathcal{X})}\} \right) + m(D) e^{-Mp/\sqrt{h(\mathcal{X})}}$$

and also goes to 0 as S is of measure 0.

To conclude, a convergence result can also be obtained for relatively irregular functions (that even do not need to be of bounded variations). The characteristic function of the von Koch flake is not of bounded variations, but its interpolants converge to it in any L_p norm as $h(\mathcal{X})$ goes to 0. To relieve the tired eyes of the reader, we display this function in figure 8.8.

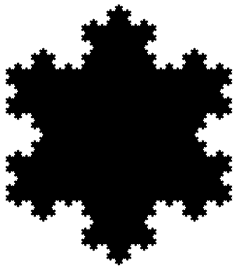


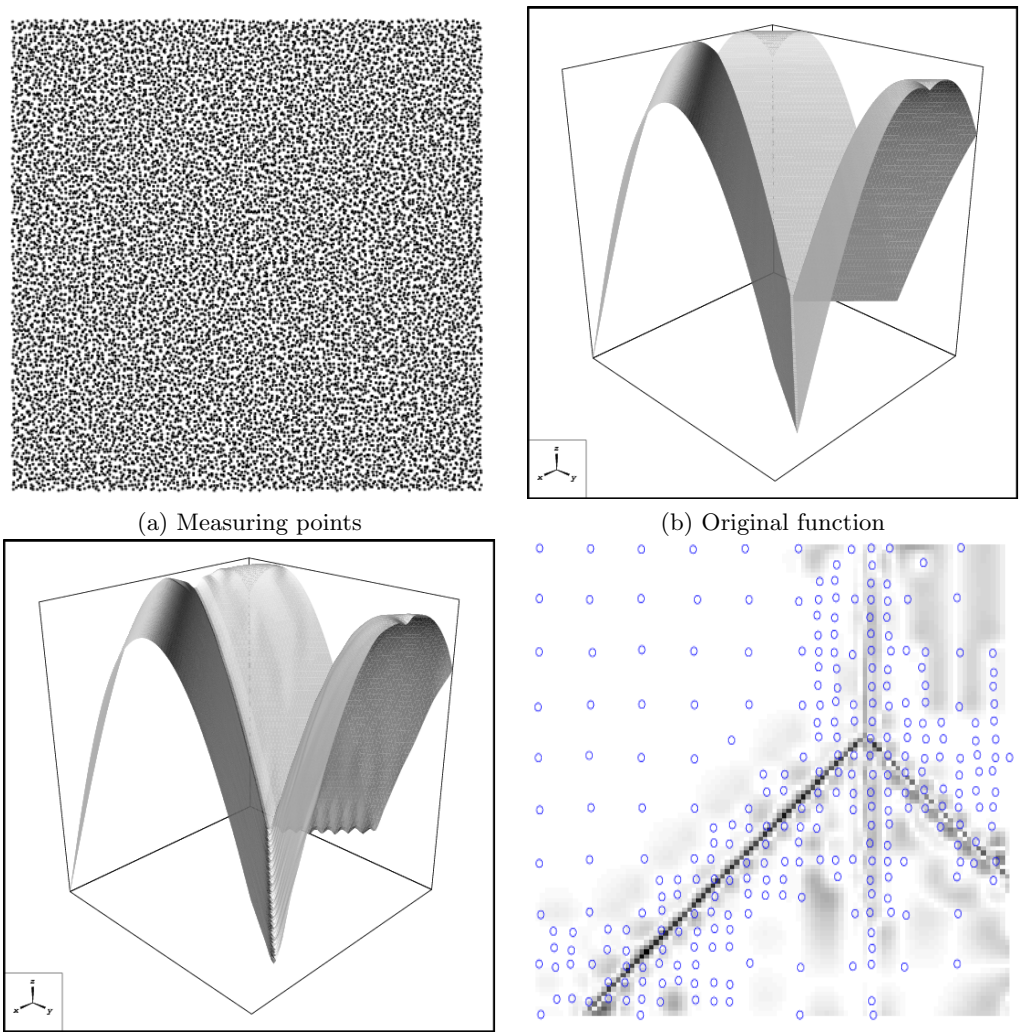
Figure 8.8: Characteristic function of the von Koch flake. Since the Hausdorff dimension of the contour is strictly larger that 1 (exactly $\log 4 - \log 3$), its length is infinite, so the function cannot be of bounded variations.

8.6 Examples

The above algorithm has been applied to samples of the piecewise smooth function

$$f(x,y) = \sin\left(3 \Big| y - |x - 0.3| - 0.4 \Big| \right)$$

The samples were $10,000$ random samples. The GRP criterion selected only 800 samples. After building the wavelet interpolant and truncation of the wavelet coefficients, only 310 coefficients (and measures) were kept. The measuring points are represented in Fig.8.9 with the original and estimated functions and the error map. The wavelet are triadic Deslauriers–Dubuc wavelets whose dual wavelets have 4 vanishing moments.



(a) Measuring points

(b) Original function

(c) Learned function

(d) Error map, and kept measures

Figure 8.9: Numerical experimentation on a sample piecewise regular function. The interpolation algorithm was applied in dimension 2. In (a), the random measuring points. In (b), the original function. In (c), the estimated function. In (d), the measuring points kept after a wavelet coefficient truncation, superimposed with the error map. We can see that the kept measures are concentrated around the function singularity.

Approximation of a discontinuous surface

The discontinuous function f defined as

$$f(x_1, x_2) = 1_{(x_1+x_2<1.2)} \left(1 - x_1^2 - x_2^2\right) + 1_{(x_1+x_2\geq 1.2)} \left(x_2^2 - 1\right)$$

was estimated on the basis of $10,000$ random samples in the square. The sampling point set $\mathcal{X}$ is displayed in Fig. 8.10. This shows that the estimation is locally adaptive, as the interpolation is more accurate where the function is smooth, and is not impaired by the existence of singularities.

8.6.1 Error decay with the number of samples

In this case, we chose a $1/2$–Lipschitz function f defined as

$$f(x) = \sqrt{\left|\sin(4x - 1)\right|} \qquad \text{for } x \in [0, 1]$$

The algorithm was then tested on increasing numbers of uniform random samples over $[0, 1]$. The L^∞ estimation error is reported in Fig.8.11. We can expect that the fill distance h is proportional to N^{-1} and thus that the error decays in $h^{1/2} \propto N^{-1/2}$.

8.7 Comments and prospects

In this section, we shortly summarize the key properties of our approach. We also suggest how to extend it to the estimation of a function on the basis of noisy measures.

8.7.1 Comparison with other methods

As compared with regularization, this methods allows to obtain a well conditioned linear system of interpolation constraints, and to achieve a convergence rate that locally depends on the function smoothness. As compared to irregular grid methods [DGS99, DR93], this method has a generic formulation that does not heavily depend on the input space dimension d.

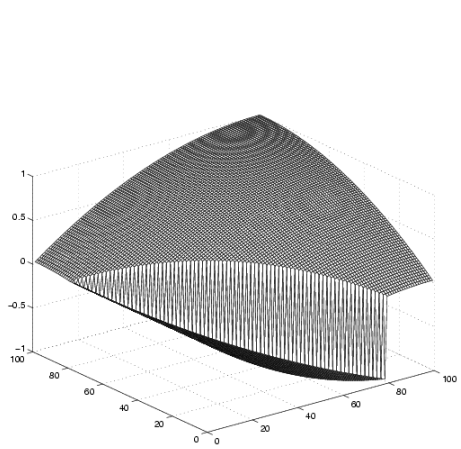
However, this method has the drawback that it does not take into account some measures. At a price that is maybe not negligible, regularization methods can use all provided measures. This will motivates some slightly different methods described in the next chapter.

8.7.2 Case of noise contaminated measures

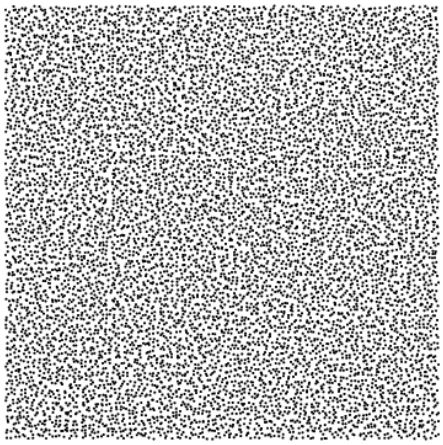
We think that this approach can be extended to estimate functions on the basis of noisy measures:

$$(\boldsymbol{x}_n, y_n = f(\boldsymbol{x}_n) + \epsilon_n)$$

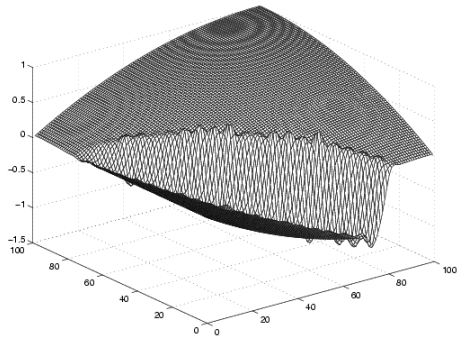
where the ϵ_n are independent realizations of a Gaussian random variable. $\mathcal{X}$ now represents the sequence $(\boldsymbol{x}_n, y_n)_n$ and $\mathcal{E}$ the sequence of measurement noises $(\epsilon_n)_n$.



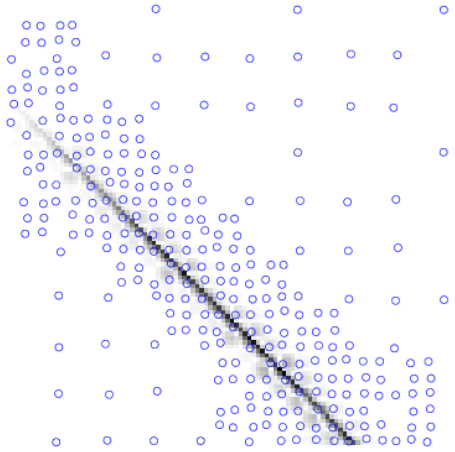
(a) Original function



(b) Sampling set $\mathcal{X}$

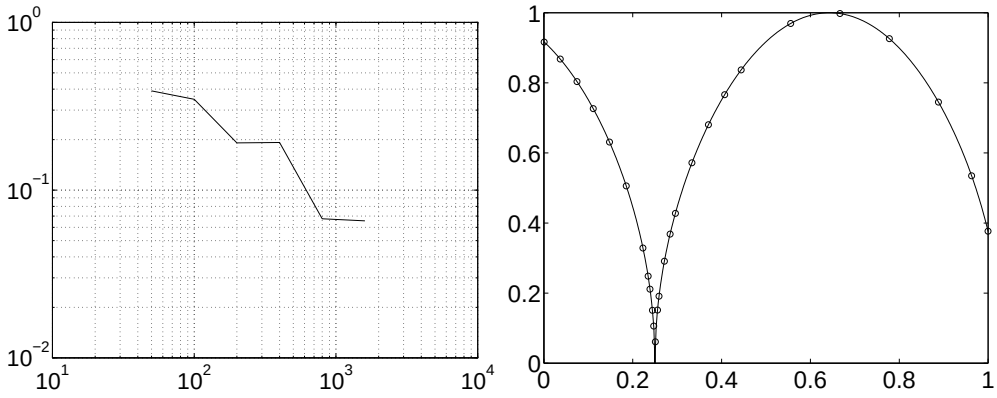


(c) Estimated function



(d) Error map and retained measures

Figure 8.10: Interpolation of a non continuous function. The main difference between the original function and the estimated function is located at the discontinuous boundary where some Gibbs effects appear in the estimated function. Note that the measures the algorithm retains to estimate the function are located close to the boundary.



(a) L^∞ error vs number of samples

(b) Original function and selected wavelet centers

Figure 8.11: Approximation of an 1/2–Lipschitz function. The left graph represents the L^∞ error decay as the number of samples increases. The decay rate is of order $N^{-1/2}$, as expected. On the right, the original function displayed together with the centers of the wavelets retained by the algorithm after truncation.

The estimation can be performed in the following way : we perform an allocation of this sequence of measures $(\boldsymbol{x}_n)_{n=1\ldots N}$ into the set of wavelet coefficients. We select a subfamily $(\boldsymbol{x}_n)_{n\in J}$ of measures and a wavelet subfamily I' according to the GRP criterion. We do not use a *set* of measures but rather a *sequence*, because it may happen that two measures at the same location provide different values y_n because of the noise.

The difference with the preceding approach consists in using all the interpolation constraints instead of a subset of them, with the selected subset of wavelets $(\phi_{j\boldsymbol{k}})_{(j,\boldsymbol{k})\in I'}$. We solve the following regression system

$$f_{\mathcal{X},\mathcal{E}} = \sum_{(j,\boldsymbol{k})\in I'} c_{j\boldsymbol{k}}\phi_{j\boldsymbol{k}}$$

$$\sum_{n=1}^{N} |f_{\mathcal{X},\mathcal{E}}(\boldsymbol{x}_n) - y_n|^2 \quad \text{is minimum.}$$

We denote in short by $A_0 c = y_0$ the system of linear constraints

$$f_{\mathcal{X},\mathcal{E}}(\boldsymbol{x}_n) = y_n \qquad \text{for } n \in J$$

and $A_s c = y_s$ the system of additional linear constraints (that would have been ignored in the preceding algorithm)

$$f_{\mathcal{X},\mathcal{E}}(\boldsymbol{x}_n) = y_n \qquad \text{for } n \notin J$$

The regression system consists in minimizing the L_2 norm of

$$\begin{bmatrix} A_0 \\ A_s \end{bmatrix} x - \begin{bmatrix} y_0 \\ y_s \end{bmatrix}$$

and the vector c is written

$$c = (A_0^T A_0 + A_s^T A_s)^{-1}(A_0^T y_0 + A_s^T y_s)$$

The preceding theorems guarantee that the square subsystem A of constraints selected by the GRP criterion is stable. The total system with all the constraints is then also stable, because $A_0^T A_0 + A_s^T A_s$ is the sum of a positive definite matrix and a positive matrix.

A second denoising step can then be done by wavelet coefficient truncation (*wavelet shrinkage*). It is however not theoretically well founded practice to do this in an interpolating wavelet basis that is not a frame of $L_2(\mathbb{R}^d)$. We have then to find a representation of the estimated function in a new basis (maybe orthogonal) where the signal is well compressed and in which the covariance matrix of the measurement noise on c, which is

$$\begin{aligned}\Gamma &= \mathrm{E}\left((A_0^T A_0 + A_s^T A_s)^{-1} \begin{bmatrix} A_0^T & A_s^T \end{bmatrix} \epsilon\epsilon^T \begin{bmatrix} A_0 \\ A_s \end{bmatrix} (A_0^T A_0 + A_s^T A_s)^{-1}\right) \\ &= (A_0^T A_0 + A_s^T A_s)^{-1}\end{aligned}$$

is close enough to a diagonal matrix.

8.7.3 Non uniform measuring point density

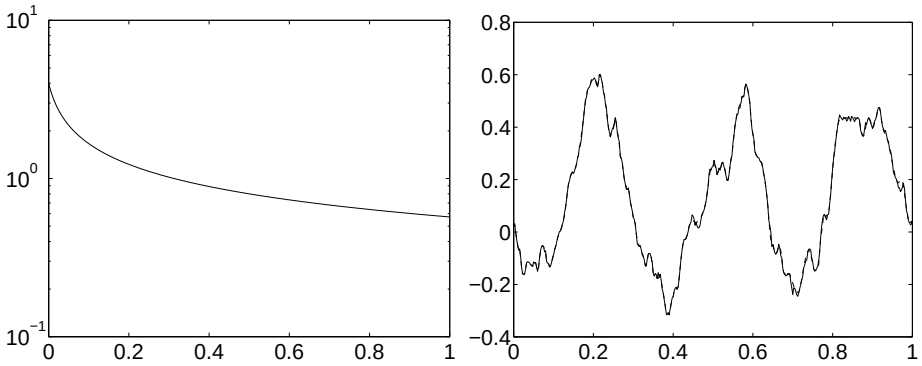
Up to now, we have always supposed to prove our theorems of convergence that the measure point density is uniformly lower bounded, i.e. that The same way as the local convergence rate of the error locally depends on the function smoothness, we can try to show that the local convergence rate also depends on the local measuring point density. We can probably expect to obtain a local or semi-local dependence on a local fill-distance of the form

$$\|f(\boldsymbol{x}) - f_{\mathcal{X}}(\boldsymbol{x})\| \leq M h(\mathcal{X}, \boldsymbol{x})^{\alpha}$$

where the fill distance is for example

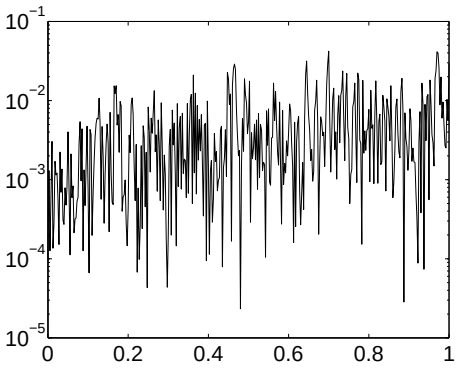
$$h(\mathcal{X}, \boldsymbol{x}) = \sup_{\substack{\boldsymbol{x}' \in D \\ |\boldsymbol{x}' - \boldsymbol{x}| < \epsilon}} \inf_{\boldsymbol{x}'' \in \mathcal{X}} |\boldsymbol{x}'' - \boldsymbol{x}'|$$

Up to now, we do not have an exact formulation of such a result nor a proof for it. However, a numerical experimentation suggests that this should be true. We have taken 1000 realizations $(x_n)_{n=1\ldots1000}$ of a non uniform random variable X. X was defined as $X = (Y + 2Y^2)/3$ where Y is a uniform random variable over $[0, 1]$. The samples were input to the algorithm to learn a Lipschitz-1/2 function. The log-density of X is given in Fig. 8.12–a, the original function (solid) and the estimated function (dashed) are represented in Fig. 8.12–b, and the resulting estimation error in Fig. 8.12–c. We see that the estimation error seems to be smaller where the sample density is higher.



(a) Drawing density

(b) Original and learned curves



(c) Estimation error

Figure 8.12: Non uniform density samples and non uniform error

Bibliography

[Can97] E. Candès. Harmonic analysis of neural nets. Technical report, Department of Statistics, Stanford University, 1997.

[Can98] E. J. Candès. Ridgelets: theory and applications. Technical Report 17, Department of Statistics, Stanford University, September 1998.

[CDDD98] A. Cohen, W. Dahmen, I. Daubechies and R. DeVore. Tree approximation and encoding. *Preprint*, 1998.

[DeV98a] R. A. DeVore. Nonlinear approximation. *Acta Numerica*, pages 51–150, 1998.

[DeV98b] R. A. DeVore. Tree approximation and encoding. In *International Workshop on CAGD and Wavelet Methods for Reconstructing Functions*. Montecatini, Italy, June 1998.

[DGS99] I. Daubechies, I. Guskov and W. Sweldens. Regularity of irregular subdivision. *Constructive Approximation*, 15(3):381–426, 1999.

[DLR90] N. Dyn, D. Levin and S. Rippa. Data dependant triangulations for piecewise linear interpolation. *IMA Journal of Numerical Analysis*, 10:137–154, 1990.

[DLR92] N. Dyn, D. Levin and S. Rippa. Boundary correction for piecewise linear interpolation defined over data-dependant triangulations. *Journal of Computational and Applied Mathematics*, 39:179–192, 1992.

[Don98a] D. L. Donoho. Orthonormal ridgelets in dimension two. In *International Workshop on CAGD and Wavelet Methods for Reconstructing Functions*. Montecatini, Italy, June 1998.

[Don98b] D. L. Donoho. Sparse components of images and optimal atomic decompositions. Technical report, Statistics Dept., Stanford University, 1998.

[DR93] N. Dyn and S. Rippa. Data-dependant triangulations for scattered data interpolation and finite element approximation. *Applied Num. Math.*, 12:89–105, 1993.

[PK93] Y. Pati and P. Krishnaprasad. Analysis and synthesis of feedforward neural networks using discrete affine wavelet transformations. *IEEE Trans. on Neural Networks*, 4(1), 1993.

[SZL+95] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P.-Y. Glorennec, H. Hjalmarsson and A. Juditsky. Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31:1691–1724, December 1995.

[ZB92] Q. Zhang and A. Benveniste. Wavelet networks. *IEEE Trans. on Neural Networks*, 3(6):889–898, November 1992.

[Zha97] Q. Zhang. Using wavelet network in nonparametric estimation. *IEEE Trans. Neural Networks*, 8(2):227–236, March 1997.

Chapter 9

Incremental methods. Prospects.

Abstract

In this chapter, we describe an incremental implementation of our approach. Such implementations can be useful in frameworks where measure samples are provided online to the interpolation network, which is also regularly quieried. In such a case, the interpolant cannot be built in a single phase, but has to be refined progressively as new samples are provided. We shortly describe an incremental version of the algorithm presented in the last chapter.

Then, we also suggest variants of this interpolation algorithm that rely on a different type of stability control: *a posteriori stability control* and *local regularization*. The main purpose of this a to avoid the picky measure selection obtained with the GRP criterion.

9.1 Incremental implementation of interpolation algorithm

The interpolation process described in the preceding chapter can be implemented in an incremental way. In order to do this, we then have to keep in memory the set of measure points, the wavelet coefficients, the (sparse) matrix of the linear system and its inverse. The iterative allocation algorithm is by definition incremental. The update of the system of interpolation constraints, of its inverse matrix and of the wavelet coefficients consists in doing low rank modifications of these matrices. These modifications can be done with the Sherman–Morrison formula.

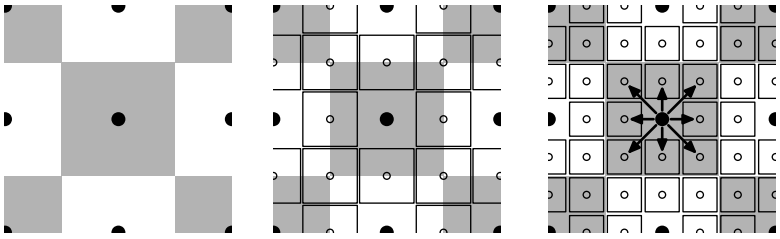
If we assume that this incremental implementation has to work online, we have to also limit the increase of the number of wavelet coefficients by removing small coefficients. This is done by removing coefficients (and the corresponding measures) that are below a given threshold. We therefore also see why in this context the allocation algorithm is useful, because it provides a reasonable linear constraint to remove together with a truncated wavelet coefficient.

The approach we describe now consists in using a tree structure in which the allocation is a tree descent. If a tree structure is not necessary to build an allocation, this structure is however easier to manage if we also have to truncate coefficients. In this case, we only truncate leaf coefficients, in order to keep the tree structure.

9.1.1 Tree structure

In order to have a tree structure for which the allocation process is a tree descent, we had to design a triadic wavelet tree structure, with triadic Deslauriers–Dubuc wavelets (as described in Sec. B.6).

The reason why we use a triadic structure is that the Voronoï[1] cells of the wavelets at a given resolution have to be embedded in the Voronoï cells of the coarser resolution wavelets. This is not the case for dyadic interpolation wavelets, as is illustrated in Fig. 9.1. If however we use triadic wavelets, the basins at resolution j are included in the basins at resolution $j-1$ and the allocation thus consists in doing a tree descent.



Basins for V_0 Basins for V_1, dyadic Basins for V_1, triadic

Figure 9.1: Why a triadic tree structure. This is illustrated in the three above graphs. The left graph represents the Voronoï cells at resolution $j = 0$. In the dyadic case, the Voronoï cells of the wavelets at resolution $j = 1$ are not embedded in those of the wavelets of resolution $j = 0$ (middle graph). In the triadic case however, this is true (right graph).

The tree structure is therefore the following one. For a domain which is $[0,1]^d$, the radices of the tree are wavelets of resolution $j = 0$ and of any translation index $\boldsymbol{k} \in \{0,1\}^d$. Then, each wavelet of indices j and $\boldsymbol{k}$ has the following sons:

$$(\phi_{j+1,3\boldsymbol{k}+\boldsymbol{\epsilon}})_{\boldsymbol{\epsilon}\in\{-1,0,1\}^d}$$

if the corresponding centers $3^{-j}\boldsymbol{k}$ are in the domain $[0,1]^d$.

This tree structure encloses all wavelets $\phi_{j\boldsymbol{k}}$ including those for which $j > 0$ and $\boldsymbol{k} \in 3\mathbb{Z}^d$. We remind that a wavelet basis is

$$\mathcal{B} = \left\{\phi_{j\boldsymbol{k}} : |\boldsymbol{k}|_\infty \leq 3^j,\ j = 0 \text{ or } \left(j > 0 \text{ and } \boldsymbol{k} \notin 3\mathbb{Z}^d\right)\right\}$$

We thus have to make the wavelets that are not in the basis (for which $j > 0$ and $\boldsymbol{k} \in 3\mathbb{Z}^d$ *silent*, that is that their coefficient is always 0. They are however in the three because they may have sons that are not silent.

In this case, the iterative allocation algorithm described in Sec. 8.3.2 can be seen as a tree descent up to a single detail. In the case when a measure falls into the basin of a silent wavelet $(j, \boldsymbol{k})$, it is pushed to the closest non silent wavelet of index $(j, \boldsymbol{k}+\boldsymbol{\epsilon})$ where $\boldsymbol{\epsilon} \in \{-1,0,1\}^d$ and $\|\boldsymbol{\epsilon}\|_1 = 1$.

[1] A Voronoï diagram of a discrete set $\mathcal{X}$ of points $\boldsymbol{x}$ is a set of Voronoï cells $B(\boldsymbol{x})$ defined by $B(\boldsymbol{x}) = \{\boldsymbol{y} \in \mathbb{R}^d : |\boldsymbol{x}-\boldsymbol{y}| < |\boldsymbol{x}'-\boldsymbol{y}|, \forall \boldsymbol{x}' \in \mathcal{X} - \{\boldsymbol{x}\}\}$.

9.1.2 Matrix updates

As is for example expressed in the Sherman-Morrison formula, changes in an inverse matrix A^{-1} resulting from low rank modifications in A can be computed in an inexpensive way. The incremental linear system inversion consists thus in permanently storing matrices A, A^{-1} the measure vector y and the coefficient vector c. To solve the linear interpolation system

$$Ac = y$$

we have to handle two cases: add a wavelet and measure point pair, and replacing a measure with another one.

- adding an interpolation constraint and a wavelet,
- the reverse operation: removing an interpolation constraint and a wavelet.

Adding a wavelet and a measure If we add a wavelet $\psi_{j\boldsymbol{k}}$ and a measurement at $\boldsymbol{x}$ so that I and $\mathcal{X}'$ are replaced with $I \cup \{(j, \boldsymbol{k})\}$ and $\mathcal{X}' \cup \{\boldsymbol{x}\}$ respectively, the corresponding matrix updates are

$$A \leftarrow \begin{bmatrix} A & V \\ L & \theta \end{bmatrix}$$

where V is the vector of coefficients $\psi_{j'\boldsymbol{k}'}(\boldsymbol{x})$ for $(j', \boldsymbol{k}') \in I$, L is the row matrix of coefficients $\psi_{j\boldsymbol{k}}(\boldsymbol{x}')$ for $\boldsymbol{x}' \in \mathcal{X}'$, and $\theta = \psi_{j\boldsymbol{k}}(\boldsymbol{x})$.

Setting $\alpha = LA^{-1}V$, A^{-1} is then updated with a Sherman–Morrison–like formula:

$$\begin{bmatrix} A & V \\ L & \theta \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} + \frac{A^{-1}VLA^{-1}}{\theta-\alpha} & -\frac{A^{-1}V}{\theta-\alpha} \\ -\frac{LA^{-1}}{\theta-\alpha} & \frac{1}{\theta-\alpha} \end{bmatrix} \tag{9.1}$$

The vector c can also be updated, with the following formula:

$$c \leftarrow \begin{bmatrix} c + \dfrac{A^{-1}V(LA - y)}{\theta - \alpha} \\ \dfrac{y - Lc}{\theta - \alpha} \end{bmatrix}$$

In this last computation, the highest complexity is carried by the multiplication $A^{-1} \times V$. One may wonder why it is not simpler to just compute again $A^{-1}y$ with the new A^{-1} and y matrices. The difference is that y has no reason to be a sparse vector, while V has.

Removing a wavelet and a measure Similar update computations can be established for the case when a measure is replaced with another one.

We assume that the inverse of the matrix

$$A = \begin{bmatrix} A' & U \\ L & \theta \end{bmatrix}$$

is known and written

$$\begin{bmatrix} B & V \\ K & \tau \end{bmatrix}$$

and that we remove the interpolation constraint corresponding to the last row and the wavelet corresponding to the last column. The new system matrix is then simply A', i.e. the update of A is written

$$A \leftarrow A'$$

We can then extract A'^{-1} from the following relations

$$\begin{aligned} B &= A'^{-1} + \frac{A'^{-1}ULA'^{-1}}{\theta - \alpha} \\ V &= -\frac{A'^{-1}U}{\theta - \alpha} \\ K &= -\frac{LA'^{-1}}{\theta - \alpha} \\ \tau &= \frac{1}{\theta - \alpha} \end{aligned}$$

We obtain

$$A'^{-1} = B - \frac{VK}{\tau} \tag{9.2}$$

The update of the coefficient vector c is also done with a similar formula. If suppose that c can be written

$$c = \begin{bmatrix} c' \\ c_1 \end{bmatrix}$$

where $c_1 \in \mathbb{R}$ and y is written

$$y = \begin{bmatrix} y' \\ y_1 \end{bmatrix} \qquad \text{where } y_1 \in \mathbb{R}$$

the vector y will simply be replaced with y'. The vector c has to undergo the following update:

$$c \leftarrow c' - \frac{V}{\tau}$$

We can establish that the matrix A and its truncated inverse are sparse, as well as the update vectors. Almost any row or column vector can be estimated to have $\mathcal{O}(j_{\max})$ non zero coefficients, except for c and y. Thus, the above computations have a complexity that is bounded by the square of $j_{\max}$. If we consider the case when the density of measure points is uniform, then j is the logarithm of the number N of measure points, and the complexity of a single iteration is thus some $\mathcal{O}(\log N)^2$, which can be considered as a real time implementation.

9.2 A posteriori stability control

The main disadvantage of the above described method is that it removes a large proportion of the provided samples, because we cannot guarantee that we can take them into account in a stable way.

A second approach consists in controlling afterwards whether adding a new measure and wavelet pair to the linear system of constraints really makes it unstable or not, and not by a sufficient geometric GRP criterion.

The matrix norm we want to control is the operator norm in $L(\ell^\infty, \ell^\infty)$ of the inverse system matrix. For a matrix B of coefficients (b_{ij}), this norm is simply:

$$\|B\|_{L(\ell^\infty,\ell^\infty)} = \sup_i \sum_j |b_{ij}|$$

The base iteration consists in keeping a copy of the current variables y, c, A and A^{-1}, to do the computations consisting in updating these variables when we have a new interpolation constraint and a new wavelet. Two cases may happen

- either the norm A^{-1} stays in an acceptable limit, and the measure and wavelet pair are kept. The provisional changes in the matrices and the coefficients are validated.
- either the norm of the matrix A^{-1} goes beyond the acceptable limit, and the measure and the wavelet are not kept. The provisional changes in the variables are canceled.

It is more difficult to give a convergence proof for such an approach, because the conditioning of the matrix A cannot be altered only because a single measure and wavelet pair induces alone instability. The cause for the bad conditioning can be diluted among all wavelet and constraint pairs. However, when no new measure is added, the allocation algorithm tends to bring the measures always closer to their corresponding wavelet centers, and thus to enhance the matrix conditioning. For this reason, we can reasonably hope that we will not get stuck in a situation where no new measure can ever be accepted by the algorithm.

9.2.1 Overall description

The principle of the a posteriori stability control is the following. For each measure, the allocation suggests a possible wavelet to add to the function representation together with the new interpolation constraint. A first update computation is done in order to estimate the norm of the update inverse system matrix. If the new norm of the inverse matrix is above a given threshold, the new measure/wavelet pair is rejected. This approach is briefly summarized in Fig. 9.2.

9.2.2 Measure replacement

If the new measure/pair is rejected, we then try to check whether the new measure cannot replace another one because it is closer to the allocated wavelet center. This is also done with the same tree descent. If this is possible, the new measure is kept, while an older is removed from the system, and no wavelet is added.

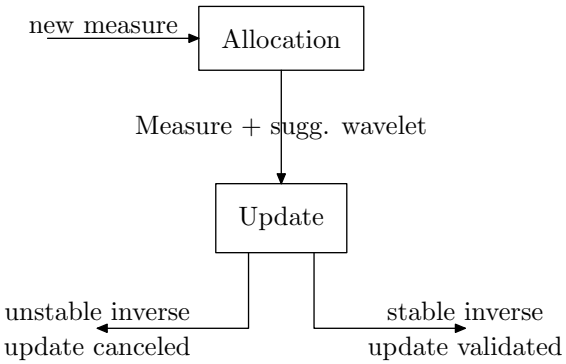


Figure 9.2: Schéma de principe du contrôle a posteriori

9.2.3 Growth control

The growth of the constructed wavelet tree is controlled by scanning the modified finest scale wavelets (i.e. the leaves of this tree). Whenever such a coefficient is beyond a truncation threshold, it is removed together with the measure it was allocated.

Wavelet shrinkage is also usually considered as an efficient process to denoise a signal of fixed size. In our case however, numerical experimentation suggests that using larger truncation thresholds to denoise measures with a Gaussian noise does not work when it is performed online. A theoretical reason is given by Donoho, because the threshold is proportional to the noise standard deviation and also increases with the number N of noise coefficients. In the case of online interpolation, the number of samples is potentially infinite, and the process then seems to keep the noise outliers of the measures.

Choice of a threshold for the stability control

Considering the stability results proved in the preceding chapter, the norm of the matrix A^{-1} is of order $\mathcal{O}(j_{\max})$ where $j_{\max}$ is the finest resolution of a wavelet in the selected wavelet subfamily. The threshold on the matrix norm we test a posteriori will therefore be of that magnitude.

Since the control of the operator norm for A^{-1} is done in the normal wavelet basis, and not in the relocated wavelet basis (see Sec. 8.5.2), the error bound is less tight. The norm of the matrix A^{-1} in the relocated wavelet basis is bounded by some multiple of $j_{\max}^2$ only.

If we now assume that we have enough measures to have allocated all wavelets up to a given resolution j, and have selected no wavelet of resolution above $j_{\max}$, we have in a similar way to the theorem 8.9, the error bound[2]

$$\|f - f_{\mathcal{X}}\|_\infty \leq M 3^{-\alpha j} j_{\max}^2$$

[2]The powers of 2 are replaced here with powers of 3 as we use triadic wavelets and not dyadic ones.

we can compare with the result of Th. 8.9:

$$\|f - f_{\mathcal{X}}\|_\infty \leq M 3^{-\alpha j}$$

The bound we obtain also supposes that we control the finest resolution $j_{\max}$ arising in the selected subfamily. Since $j_{\max} \geq j$, the error decay rate is slightly lower than that obtained in the preceding chapter. To have an estimation error going to 0, we have to require at least a Lipschitz regularity for some positive exponent $\alpha > 0$, while the former method was able to approximate successfully only continuous functions.

We can therefore at most show a result of *conditional convergence*. Assuming that we can select wavelets up to any given scale without deteriorating the system matrix conditioning, and without selecting coefficients of too high a resolution $j_{\max}$, we can guarantee that the estimated function can be arbitrarily close to the original one, provided that the original function is at least Lipschitz–α for some $\alpha > 0$.

9.3 Partial regularization

We have to notice that this new approach as such also removes measures it cannot handle without impairing the system stability. We also defined another approach, of *partial regularization*,that aims at providing an interpolant without rejecting any measure.

Let us go back to the a posteriori stability control described above. As the allocation scheme submits a new measure/wavelet pair, the a posteriori stability control keeps the measure if the new system is stable enough and rejects it otherwise.

The alternative we have started to develop and to implement consists in adding new wavelets called regularization wavelets, and whose centers are located close to the new measure. Their purpose is to provide a way to explain the new measure that is not in contradiction with the former measures. Since we then obtain a linear system with more wavelets than constraints, we have to solve this system with for example a least square method, i.e. we have to find the minimum

$$\min \left\{ \|c\|_H : \begin{bmatrix} A_0 & A_r \end{bmatrix} c = y \right\}$$

where A_0 is the submatrix of the system of interpolation constraints that corresponds to the wavelets selected by allocation, and A_r is the submatrix corresponding to regularization wavelets. We define

$$A = \begin{bmatrix} A_0 & A_r \end{bmatrix}$$

In a similar way to what happens in regularization, the choice of a given Hilbert metric $\|.\|_H$ corresponds to choosing a given Sobolev norm, and has a crucial influence on the result.

We have considered building this metric in an adaptive way by defining a weight $\gamma_{j\boldsymbol{k}}$ for each coefficient $c_{j\boldsymbol{k}}$. For each wavelet coefficient $c_{j\boldsymbol{k}}$ to which a measure is allocated, we set $\gamma_{j\boldsymbol{k}} = 1$. For other coefficients (regularization coefficients), we set[3] $\gamma_{j\boldsymbol{k}} = 3^{-d\alpha}\gamma_{j-1,\boldsymbol{k}'}$ where

[3] This formula holds for triadic wavelets.

the wavelet $\phi_{j-1\boldsymbol{k}'}$ is the parent of $\phi_{j\boldsymbol{k}}$. The Hilbert metric on the wavelet coefficients is then:

$$\|c\|_H^2 = \sum_{j\boldsymbol{k}} \frac{c_{j\boldsymbol{k}}^2}{\gamma_{j\boldsymbol{k}}^2}$$

This approach can also be implemented incrementally. If we denote by γ the diagonal matrix of coefficients

$$\Gamma_{(j,\boldsymbol{k})(j',\boldsymbol{k}')} = \delta_{jj'}\delta_{\boldsymbol{k}\boldsymbol{k}'}\frac{1}{\gamma_{j\boldsymbol{k}'}^2}$$

the solution of the regularized linear system is

$$c = \Gamma^{-1}A^T\left(A\Gamma^{-1}A^T\right)^{-1}y$$

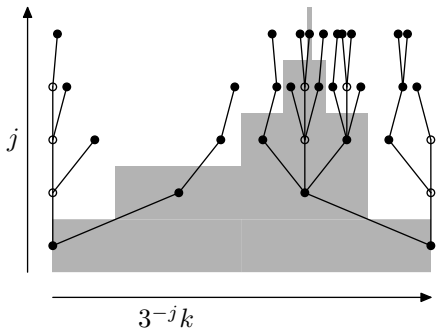


Figure 9.3: Tree example with two categories of wavelets. The solid circles represent wavelet coefficients. Wavelets on a grey background are wavelets to which a measure is allocated; their weight $\gamma_{j\boldsymbol{k}}$ is always 1. Wavelets on a white background are regularization wavelets; their weight is at most $3^{-\alpha}$ times that of their parent. We remind that empty circles represent silent wavelets, whose coefficient is always 0.

Introducing the matrix Γ does not considerably increase the complexity of the computations, since it is diagonal. However, we have to solve a least square problem in a incremental way, and this supposes to store the current values of A, $A\Gamma^{-1}A^T$ and its inverse $\left(A\Gamma^{-1}A^T\right)^{-1}$.

9.3.1 Numerical experiments and prospects

The two approaches of a posteriori selection and of local regularization have been implemented. In the first, the stability control has however been implemented in a weaker way. Instead of verifying explicitly the operator norm of A^{-1}, the test was based on the sole parameter $\theta - \alpha$ that appears in the update formula (9.1). We only accept measure/wavelet

pairs for which $|\theta - \alpha|$ is larger than some threshold. We know that the new systems we build are then non degenerate. However, the upper bound we obtain from such a selection rule increases and diverges as the number of updates increases. In practice, if there is no drastic explosion of the numerical results, the provided estimation cannot be considered as stable.

These implementations have been tested to estimated a function in a setting where the algorithm based on the GRP criterion fails, because it assumes that the measuring points asymptotically fill the whole definition domain of the function. In this test case however, 200 measuring points are draw randomly with a uniform density on the circle. In this case, the function profile is close to the function to estimate, but instabilities happen to appear, and this is all the worse as the number of samples is large. The results are shown on figure 9.4.

These results show that we can get rid of the constraint that the measure point density should not have any holes. However, the stability of the current algorithm is not yet satisfying. A second set of examples demonstrates it clearly. The function to estimate is a piecewise smooth function of a single variable

$$f(x) = x + \sin 6x \times 1_{(x>1/2)}$$

The result of the approximation by GRP measure selection and by local regularization are reported on figure 9.5. Obviously the local regularization method does not yet provide satisfying results, especially compared to the GRP method. The result with our implementation of an a priori stability control is also unstable, for reasons we already know.

For the first approach, we have already established why the our results are unstable: the norm of the inverse matrix A^{-1} is not bounded uniformly in time, but only by a sharply increasing sequence. This can be implemented in a stable way only by effectively checking that the operator norm of the inverse matrix does not exceed a fixed threshold. The operator norm of the system inverse matrix can be estimated in an incremental way.

For local regularization, we can suggest the following heuristic analysis. This approach is midway between a classical regularization (for which the weights $\gamma_{j\boldsymbol{k}}$ are defined as $\gamma_{j\boldsymbol{k}} = 3^{-\alpha dj}$), and an unstable square system obtained by allocation, we can get by removing all regularization wavelets from the expansion of $f_{\mathcal{X}}$:

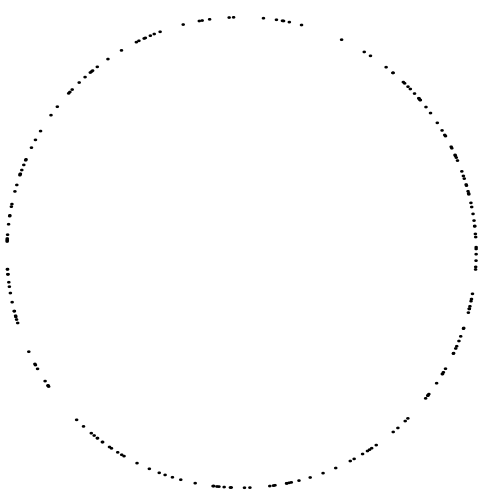
$$f_{\mathcal{X}} = \sum_{\gamma_{j\boldsymbol{k}}=1} c_{j\boldsymbol{k}} \phi_{j\boldsymbol{k}}$$

to get back to a square linear system:

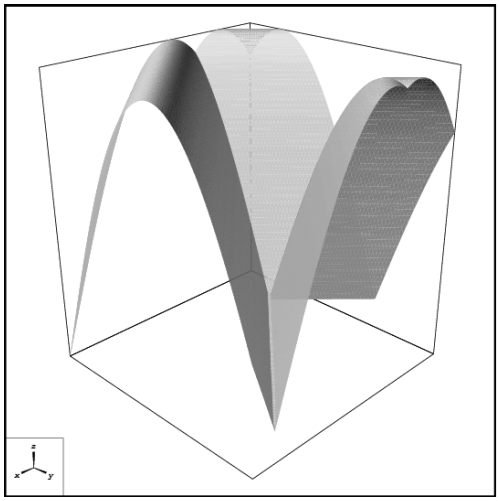
$$A_0 c_0 = y$$

If however a weight of 1 is only given to wavelets of a subtree that make up a stable system of interpolation constraints, selection either with the GRP criterion or with an a posteriori stability control, we can hope to get a stable least square system. We see that the study of this local regularization approach will benefit from the knowledge of the two first approaches (GRP and a posteriori stability control).

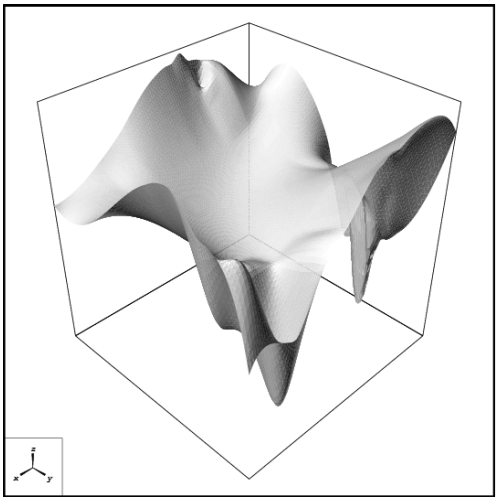
The advantage we are hoping to gain from this approach of local regularization is to limit the increase of the condition number of the collocation matrix. In order to achieve



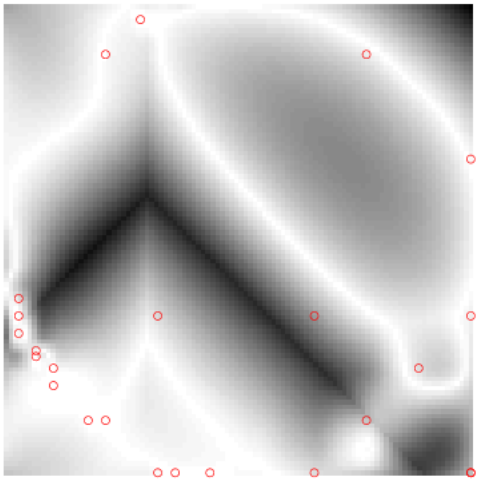
(a) Measuring points



(b) Original



(c) Learned



(d) Error map and wavelet centers

Figure 9.4: Function learning over a submanifold: a circle. The sampling points are represented in (a). In (b), the original function; in (c) the learned function (local regularization); in (d) the error map and the remaining wavelet coefficient centers.

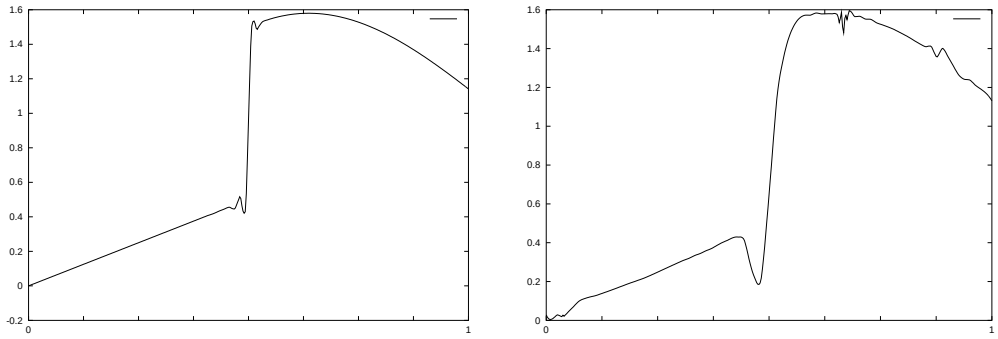


Figure 9.5: Approximation of a piecewise smooth function on the basis of 200 samples. On the left, the approximation algorithm based on the GRP criterion, and right the local regularization.

this, we avoid doing regularization for wavelet coefficient that do not require it, i.e. for those involved in a square subsystem of constraints and wavelets for which a stability criterion is fulfilled.

Appendix B

Interpolating wavelets

Abstract

The literature on wavelets provides a number of different types of wavelet families, orthogonal or biorthogonal, symmetric or not, of varying smoothness. In this chapter, we examine the properties of a special kind of wavelets: interpolating wavelets. We briefly show that classical interpolation tools can be fitted into a multiresolution framework. In this, the work of Deslauriers and Dubuc in the beginning of the eighties plays a central role.

The term "interpolating wavelets" may be misleading. In an interpolating wavelet basis, the wavelet is not an interpolating function. Only the scaling function is.

B.1 Interpolating scaling functions

An interpolating scaling function is a function ϕ that is interpolating in a classical sense, namely:

$$\begin{aligned} &\phi(k) = 0 && \text{if } k \in \mathbb{Z} \text{ and } k \neq 0 \\ &\phi(0) = 1 \end{aligned}$$

and is at least continuous.

If the function ϕ is built in a multiresolution analysis as infinite convolution of discrete filters:

$$\hat{\phi}(\omega) = \prod_{k=1}^{+\infty} m_0\left(\frac{\omega}{2^k}\right)$$

the interpolating property can be expressed as the following assumption on the filter m_0:

$$m_0(\omega) + m_0(\omega + \pi) = 1 \qquad \forall \omega \in \mathbb{R}$$

Several families of such interpolating scaling functions are known (Spline functions, Deslauriers-Dubuc fundamental functions). We know since the work of Fix and Strang that an interpolating function of fixed compact support cannot have an arbitrarily high approximation order.

B.2 Interpolating wavelets

Such properties of the scaling function do not have a natural extension on the wavelet, as has the orthogonality condition. Therefore, some author just suggest to define a wavelet as a Riesz basis generating function of the orthogonal complement of V_0 in V_1, to have thus

$$W_j = V_{j+1} \cap V_j^{\perp}$$

For our own, we rather follow the approach suggested by Donoho, who is maybe inspired by the example of the Schauder basis to simply set

$$\psi(x) = \phi(2x - 1)$$

Since a convenient norm for such a family is the L_∞ norm, the dilated and translated function are defined with the normalization $\phi_{jk}(t) = \phi(2^j t - k)$, and the basis constructed by Donoho is therefore

$$\mathcal{B}_J = \{\phi_{jk} : j = J \text{ and } k \in \mathbb{Z} \text{ or else } j > J \text{ and } k \text{ is odd}\}$$

The advantage of such a choice appears very soon: the dual scaling functions are Dirac masses and the dual wavelets finite linear combination thereof. We do not prove it here, we only give an informal argument towards this. The high pass filter corresponding to Donoho's choice is:

$$m_1(\omega) = \frac{1}{2} e^{-i\omega}$$

The corresponding transfer matrix is then:

$$M(\omega) = \begin{bmatrix} m_0(\omega) & m_0(\omega + \pi) \\ e^{-i\omega}/2 & -e^{-i\omega}/2 \end{bmatrix}$$

Since the filter m_0 is an interpolating filter, the determinant $M(\omega)$ as the following simple form:

$$\det M(\omega) = -\frac{1}{2} e^{-i\omega}$$

We derive from this an expression of the dual transfer matrix $\tilde{M}(\omega) = M(\omega)^{-T}$:

$$\tilde{M}(\omega) = \begin{bmatrix} 1 & 1 \\ 2m_0(\omega + \pi)e^{-i\omega} & 2m_0(\omega)e^{-i(\omega+\pi)} \end{bmatrix}$$

We thus obtain the dual wavelets

$$\begin{aligned} \tilde{\phi}(t) &= \delta(t) \\ \tilde{\psi}(t) &= \sum_{k \in \mathbb{Z}} (-1)^k m_0[k] \delta\left(t - \frac{k+1}{2}\right) \end{aligned}$$

B.3 Deslauriers-Dubuc wavelets

The Deslauriers-Dubuc scaling functions are interpolating functions that fulfill a two scale refinement equation and are an optimal trade-off between support size and approximation order, in a similar way to the Daubechies wavelets.

More precisely, for a given even number of vanishing moments $2p$, there exists a unique filter m_0 of support size less or equal to $4p-1$ that is an interpolating filter. One can prove this by building the linear system of associated constraints, which happens to be a van der Monde system. The filters and scaling functions have been named after Deslauriers and Dubuc who discovered them in the iterative dyadic interpolation scheme. Scaling functions for several parameters p are displayed in Fig. B.1.

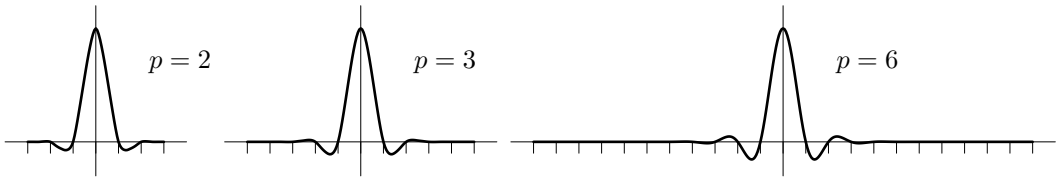


Figure B.1: Some scaling Deslauriers-Dubuc scaling functions drawn over their support.

We can also check that the autocorrelation function

$$\Phi(t) = \int_{\mathbb{R}} \phi(u)\overline{\phi(u+t)}du$$

of an orthogonal scaling function is an interpolating scaling function. Even more, the autocorrelation of a Daubechies scaling function is a Deslauriers-Dubuc scaling function.

Unlike the Daubechies filters, the Deslauriers-Dubuc filters can be computed explicitly, and their coefficients are dyadic rationals. For a given parameter p, the coefficients of the m_0 filter are the following:

$$\begin{aligned}
m_0[k] &= 0 && \text{if } |k| \geq 2p \\
m_0[2k] &= 0 && \text{if } k \neq 0 \\
m_0[0] &= \frac{1}{2} && \\
m_0[\pm(2k-1)] &= \frac{(-1)^{k+1}(2p)!^2}{2^{4p}p!^2(p-k)!(p+k-1)!(2k-1)} && \text{for } k \leq p
\end{aligned}$$

B.4 Wavelets on the interval

Most often, the support of a signal is bounded, because we only have access to some restriction to a finite interval of a larger signal. Constructing a family of wavelets to analyze a finite support signal is traditionally done in two possible ways: periodization or an explicit construction of interval wavelets with special boundary wavelets.

B.4.1 Periodization

A function defined over the whole real line $\mathbb{R}$ can be periodized into a function defined over $\mathbb{R}/N\mathbb{Z}$ with the following definition.

$$\begin{aligned} \mathcal{P} : L_1(\mathbb{R}) &\to L_1(\mathbb{R}/N\mathbb{Z}) \\ f &\mapsto (x \mapsto \sum_{y \equiv x(N)} f(y)) \end{aligned}$$

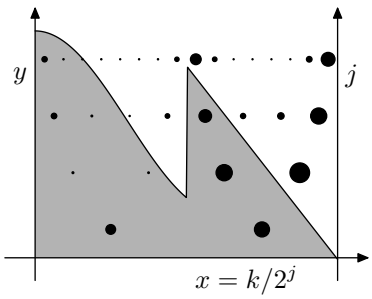


Figure B.2: Periodized wavelet decomposition of a function defined over an interval. We can see that because the function has not the same values on both ends of the interval, this is perceived as a singularity by the periodized wavelets, and we see the corresponding cone of coefficients close to the boundary.

By periodizing orthogonal wavelets that have a sufficiently fast decay, we obtain a set of function that is an orthonormal basis of $L_2(\mathbb{R}/N\mathbb{Z})$. The default of such an approach is that a smooth function defined over $[0, N]$ will have expansion coefficients of a smooth function only if it is smooth as a function defined over $\mathbb{R}/N\mathbb{Z}$, which means that it must have the same values (and same derivatives, etc.) at both ends of the interval $[0, N]$

$$\begin{aligned} f(0) &= f(N) \\ f'(0) &= f'(N) \\ &\vdots \quad \vdots \\ f^{(k)}(0) &= f^{(k)}(N) \end{aligned}$$

This is illustrated in Fig. B.2.

B.4.2 Boundary wavelets and wavelet bases on the interval

Cohen, Daubechies, Jawerth and Vial have suggested a way to build orthogonal wavelet bases of $L_2([0, N])$. With such bases, any smooth function over $[0, N]$ has fast decaying wavelet coefficients, whatever the boundary values are. An example of such an expansion is displayed in Fig. B.3.

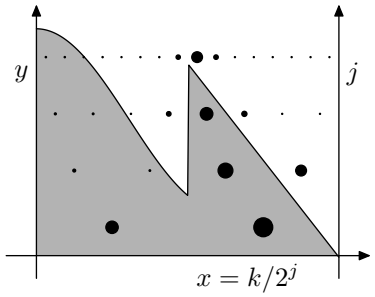


Figure B.3: Expansion of a function over a wavelet basis of the interval. We can see that unlike the case of Fig. B.2, the function does not have to be extendable to a continuous periodic function to have a fast decaying wavelet expansion.

The construction of such a basis relies on the explicit construction of boundary wavelets as linear combinations of wavelets truncated at the boundary. The resulting wavelets are always well localized, and the support of a small scale wavelet cannot contain both edges of the interval. This construction is most satisfying than periodization, but it is more complex, with the definition of edge filters and has been defined for Daubechies wavelets, and is not easy to extend to any kind of wavelet.

B.4.3 Interpolating wavelets on the interval

For interpolating wavelets, Donoho suggested an approach that is inspired from the work of Cohen, Daubechies *et al.* In this work however, we follow a simpler and more intuitive approach that consists in going back to the original approach of Deslauriers and Dubuc and adapt it for our purpose.

Back to Deslauriers and Dubuc's scheme

Deslauriers and Dubuc's dyadic refinement scheme is the following. If a function f is known over a regular grid, like the points in $\mathbb{Z}$, we construct an estimation of f over a twice finer grid with centered Lagrange interpolations. On each integer point, the estimate is unchanged. For each half integer $k + 1/2$, we denote by $\Pi_{[k,k+1]}$ the Lagrange interpolating polynomial that coincides with f on each of the points $k - p + 1, \ldots, k$ and $k + 1, \ldots, k + p$. Then, we set

$$f(k + 1/2) = \Pi_{[k,k+1]}(k + 1/2)$$

An example of such an interpolation is represented in Fig. B.4. This refinement step can be iterated indefinitely to obtain a definition of f over all dyadic rationals (i.e. of the form $k/2^j$). If the original sequence $(f(k))_{k\in\mathbb{Z}}$ is uniformly continuous (i.e. $(f(k) - f(k + 1))_{k\in\mathbb{Z}}$ is bounded), then the interpolated f over all dyadic rationals is also uniformly continuous

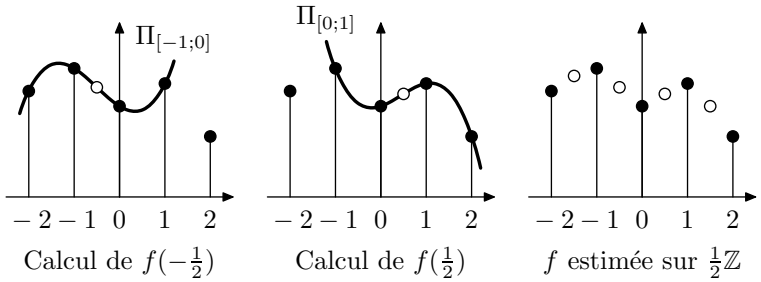


Figure B.4: Deslauriers-Dubuc dyadic refinement scheme. The black dots represent values of f on the integer grid. The white dots represent the interpolated values on the half integer grid. Two Lagrange polynomials are also drawn. In this example, their degree is 3. The two left pictures indicate how $f(-1/2)$ and $f(1/2)$ are estimated, and the third picture represents the resulting values of f over the finer grid $\mathbb{Z}/2$.

and can be extended by continuity over $\mathbb{R}$[1].

The dyadic interpolation process is linear, and the intermediate values can be obtained by filtering:

$$f(k+1/2) = \sum_{l\in\mathbb{Z}} m_0[2l+1]f(k-l)$$

where m_0 is the Deslauriers-Dubuc interpolating filter.

Thus the construction of ϕ_{jk} can be done as follows. We start from the grid $\mathbb{Z}/2^j$ on which we define ϕ_{jk} by:

$$\phi_{jk}\left(\frac{l}{2^j}\right) = \begin{cases} 0 & \text{if } k \neq l \\ 1 & \text{if } k = l \end{cases}$$

We extend this definition over all $\mathcal{D}$ by iterating the dyadic refinement scheme, and to all $\mathbb{R}$ by density. Wavelets can then defined from such scaling functions by setting simply $\psi_{jk} = \phi_{j+1,2k+1}$.

Adapting the scheme to the interval

This construction can be adapted to the interval. The problem is to define the refinement scheme close to the boundaries of the interval. If we start from an integer grid of finite support $[0, N]$ and if the Lagrange polynomials are of degree 3, the classical Deslauriers-Dubuc refinement scheme cannot be used to estimate $f(1/2)$ or $f(N-1/2)$. Indeed, the estimation of $f(1/2)$ would rely on samples of f on -1, 0, 1 and 2, but f is only known over $\{0, \ldots, N\}$. We circumvent this difficulty by using a shifted Lagrange polynomial. $f(1/2)$ is estimated with the polynomial $\Pi_{[1;2]}$ that is well defined. Thus, the same polynomial is used to interpolate f on $1/2$ and $3/2$.

[1]In fact, since the resulting fundamental interpolating function is compactly supported, the uniform continuity assumption is not necessary in order to have a continuous interpolant.

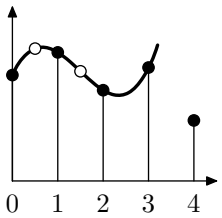


Figure B.5: Shifted interpolation. The same polynomial is used to estimate $f(1/2)$ and $f(3/2)$.

In general, for a given even interpolation order $2p$, we can adapt the Deslauriers-Dubuc to the interval. This requires $p-1$ boundary filters in addition to the classical Deslauriers-Dubuc filter. We can show that the scaling functions built this way have the same smoothness as those defined over the whole interval. At each scale, we have $2p$ boundary wavelets at each side of the interval:

$$\begin{cases} \phi^g_{jk} & \text{for } 0 \leq k < 2p \text{ on the left} \\ \phi^d_{jk} & \text{for } N2^j - 2p < k \leq N2^j, \text{ on the right} \\ \phi_{jk} & \text{for } 2p \leq k \leq N2^j - 2p, \text{ unchanged} \end{cases}$$

The boundary wavelets ϕ^g_{jk} and ϕ^d_{jk} the wavelets ϕ_{jk} whose support is not in $[0, N]$.

The scale spaces $V-J$ on the interval are spanned by the $N2^j+1$ scaling function we can build this way. A detail space W_j can then be built by subsampling V_{j+1}, i.e. by selecting from the $\phi^g_{j+1,k}$, $\phi_{j+1,k}$ and $\phi^d_{j+1,k}$ the wavelets of odd index k.

Remarks

- *It is worth noticing that the natural dimension of a scale space V_j is not some multiple of 2^j any more, but rather some "multiple of 2^j plus 1".*
- *The smoothness is as said at least the same as that of the wavelets defined over the real line. The reason is that inside the interval, they coincide with linear combinations of translates of ϕ, and on some vicinity of the boundaries they are polynomial. Indeed, the wavelets ϕ^g_{jk} or ϕ^d_{jk} are polynomial on respectively $[0, 2^{-j}]$ and $[N - 2^{-j}, N]$.*

Other constructions

Such a construction has a drawback it share with the orthogonal wavelets on the interval by Cohen, Daubechies, Jawerth and Vial. The boundary wavelets have an overshoot of amplitude increasing with the parameter p, while wavelets over the real line are bounded by 1.

Indeed,

- for $p = 2$, $\max |\phi^g_{0,1}| > 1.04$

- for $p = 3$, $\max |\phi^g_{0,1}| > 1.25$
- for $p = 6$, $\max |\phi^g_{0,3}| > 6.8$
- for $p = 12$, $\max |\phi^g_{0,5}| > 665$

A way avoid such explosions consists in taking boundary filters of lower order, with then impairs the coefficient decay on the boundary.

In another case, such a tactic is useful: if the scale is so large that the real line wavelets overlap both sides of the interval. If for example the considered interval is $[0, 1]$, the integer grid only has two points. It is not possible to use a Lagrange interpolation scheme of order higher that 1. Turning to lower order filters for the lowest resolutions is thus a way to build wavelet bases of arbitrary order with a scale V_0 of dimension 2 (which is the lowest possible).

To build a Deslauriers-Dubuc wavelet basis of $\mathcal{C}([0, 1])$, the refinement from resolution $j = 0$ to resolution $j = 1$ uses a Lagrange polynomial of degree 1, the refinement from $j = 1$ to $j = 2$ Lagrange polynomials of degree 2. The following refinements use the Deslauriers-Dubuc for the interval as described above. In Fig. B.6, the refinements are represented on the left and the resulting wavelets on the right.

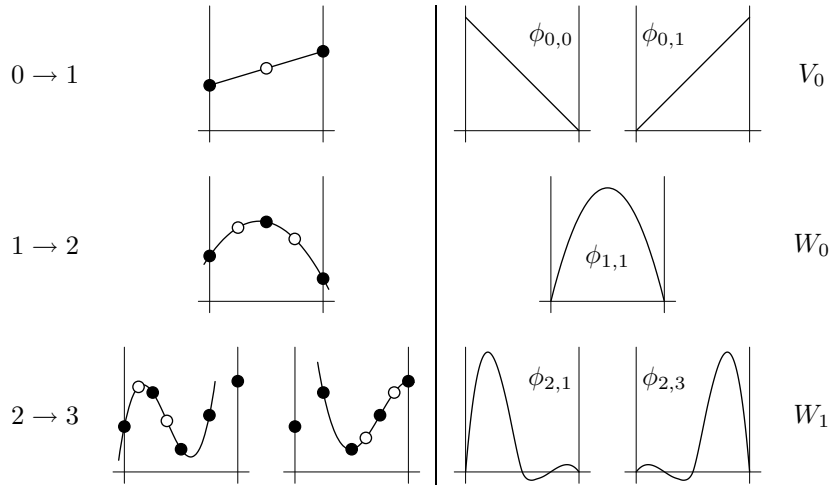


Figure B.6: Low resolution dyadic refinement (on the left) and resulting wavelets (on the right), for an interpolating wavelet basis with 4 vanishing moments.

B.5 Uniform approximation theorems

In most of the cases, the metric used to measure convergence is the L_2 metric. This metric is fairly natural and convenient in signal processing problems.

For interpolating wavelets however, with Donoho's choice of detail spaces, the most natural metric is the L_∞ one, most notably because the dual wavelets are finite sums of Dirac masses and are not square integrable.

A first theorem has been proved by Donoho, who shows that a uniformly continuous function can be approximated with its projection on V_j with an arbitrary precision, whenever j is large enough. Donoho assumes that the interpolating scaling function has to be of exponential decay.

Theorem B.1 (Donoho)

If $f : \mathbb{R} \to \mathbb{R}$ is uniformly continuous, and if the interpolating function ϕ has an exponential decay, then

$$\sup_{t\in\mathbb{R}} \left| f(t) - \sum_{k\in\mathbb{Z}} f\left(\frac{k}{2^j}\right) \phi(2^j t - k) \right| \tag{B.1}$$

is finite and goes to 0 as $j \to +\infty$.

We can suggest some variations on this theorem. Either we keep the hypothesis that f is uniformly continuous. In this case, the decay we assume for ϕ has to be sufficient to guarantee that the sums

$$\sum_{k\in\mathbb{Z}} f\left(\frac{k}{2^j}\right) \phi(2^j t - k) \tag{B.2}$$

be uniformly convergent in t. Since a uniformly continuous function increases at most linearly with t, we can relax the decay assumption on ϕ to

$$\exists \epsilon > 0,\ M > 0 : |\phi(t)| < \frac{M}{1 + |t|^{2+\epsilon}}$$

which is a much lower decay rate.

Another approach consists in reducing the assumption on the function f, by only assuming that it is continuous. In this case, we have to assume that ϕ is compactly supported, to guarantee that the sums (B.2) are well defined.

Theorem B.2

If $f : \mathbb{R} \to \mathbb{R}$ is continuous, and if the interpolating function ϕ is of compact support, then

$$\sup_{t\in[-M,M]} \left| f(t) - \sum_{k\in\mathbb{Z}} f\left(\frac{k}{2^j}\right) \phi(2^j t - k) \right|$$

goes to 0 as $j \to +\infty$, pour any M. In other words, the approximation of a continuous function by sums of interpolating functions converges to f for any f in $C(\mathbb{R})$, with the Fréchet family of seminorms $(\mathcal{N}_n)_{n\in\mathbb{N}^*}$:

$$\mathcal{N}_n(f) = \sup_{t\in[-N,N]} |f(t)|$$

We can state a similar theorem for interval decompositions, where there is distinction between continuous and uniformly continuous functions. Moreover, if we make stronger assumptions on the function f, we can bound explicitly the approximation error (B.1). If f is α–Lipschitz, the approximation error can be bounded by some $M2^{-\alpha j}$.

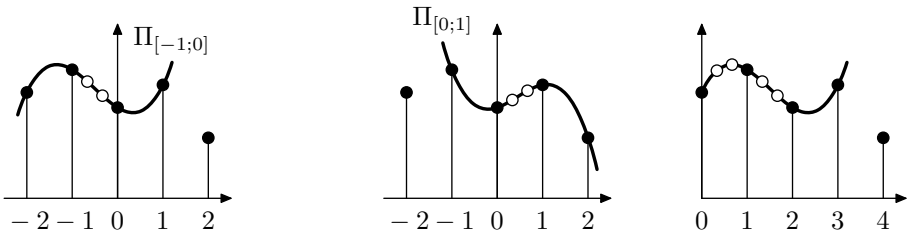
B.6 Triadic Deslauriers-Dubuc wavelets

In some variants of the learning algorithms described in the previous chapters, we need interpolating wavelets that are triadic and not dyadic. Such wavelets can easily be built by adapting the above dyadic construction scheme. Instead of a dyadic function refinement from the grid $\mathbb{Z}$ to the grid $2\mathbb{Z}$, we design a triadic refinement to interpolate a function defined on $\mathbb{Z}$ to a function defined on $3\mathbb{Z}$.

This is again done with a moving Lagrange interpolation. For each integer interval $[k, k+1]$, we pick symmetric integer support points $k-p+1, \ldots, k$ and $k+1, \ldots, k+p$, fit a polynomial $\Pi_{[k,k+1]}$ of lowest possible degree to the function f on these points, and set

$$f(k+1/3) = \Pi_{[k,k+1]}(k+1/3)$$
$$f(k+2/3) = \Pi_{[k,k+1]}(k+2/3)$$

This is illustrated in Fig. B.7–a,b. This scheme can also as before be adapted to a finite interval. Again, the interpolation mask is shifted at the boundaries so that the reference points always are in the interval, as is shown in Fig. B.7–c.



(a) Comput. of $f(-\frac{2}{3})$ and $f(-\frac{1}{3})$ (b) Comput. of $f(\frac{1}{3})$ and $f(\frac{2}{3})$ (c) Shifted mask

Figure B.7: Triadic refinement scheme

For Lagrange polynomials of degree 3, which corresponds to a choice of a parameter $p = 2$, we have again an interpolation filter

$$m_0(\omega) = \frac{81 + 120\cos\omega + 60\cos 2\omega - 10\cos 4\omega - 8\cos 5\omega}{243} \tag{B.3}$$

that define by infinite convolutions an interpolating function

$$\hat{\phi}(\omega) = \prod_{j=1}^{+\infty} m_0\left(\frac{\omega}{3^j}\right)$$

If we define the wavelets ϕ_{jk} to be $\phi(3^j x - k)$ on the real line and use a similar device as for dyadic wavelets on the interval, we can exhibit new bases

$$\{\phi_{jk} : (j > 0 \text{ and } 3k \notin \mathbb{Z}) \text{ or } j = 0\}$$

for the real line, and

$$\left\{\phi_{jk} : 0 \leq k \leq N3^j, (j > 0 \text{ and } 3k \notin \mathbb{Z}) \text{ or } j = 0\right\}$$

for the interval $[0, N]$.

The triadic wavelet tunnel effect

A noteworthy difference between dyadic and triadic wavelets appears in the construction of interval functions. Boundary wavelets, that have a different shape from wavelets of the whole real line occur at locations in the interval where precisely a wavelet of the whole real line would also fit into the interval. This is related to the fact that the wavelet support size s_w is in the triadic case strictly smaller than the filter support size s_f because

$$s_w = \left(\frac{1}{2} + \frac{1}{4} + \dots\right) s_f = s_f \qquad \text{in the dyadic case,}$$

$$s_w = \left(\frac{1}{3} + \frac{1}{9} + \dots\right) s_f = \frac{1}{2} s_f \qquad \text{in the triadic case.}$$

This is different from the case of dyadic wavelet where usually special boundary wavelets are built to replace normal wavelets that would overlap the boundary.

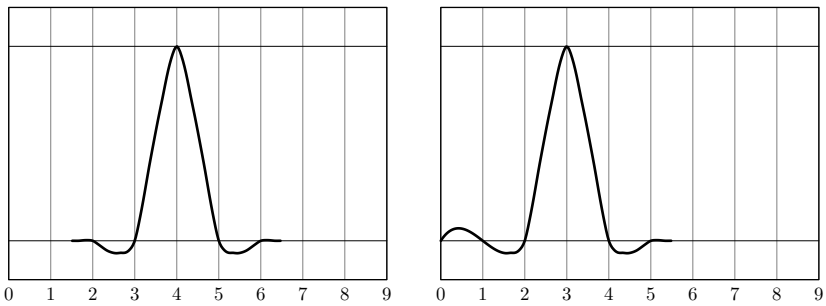


Figure B.8: Triadic wavelets on the interval. In (a), the wavelet ϕ_{04} in the interval $[0, 9]$ is not a boundary wavelet, and has the support $[1.5; 6.5]$. In (b), the wavelet ϕ_{03} is a boundary wavelet, has support $[0; 5.5]$. What occurs in the triadic case is that the wavelet ϕ_{04} shifted to the left would have fitted into the interval, but the corresponding wavelet is however a boundary wavelet.

Bibliography

[CDV93] A. Cohen, I. Daubechies and P. Vial. Wavelets on the interval and fast wavelet transforms. *Applied and Computational Harmonic Analysis*, 1(1):54–81, 1993.

[DD87] G. Deslauriers and S. Dubuc. Interpolation dyadique. In *Fractals, dimensions non-entières et applications*, pages 44–55. Masson, Paris, 1987.

[DD89] G. Deslauriers and S. Dubuc. Symmetric iterative interpolation processes. *Constructive Approximation*, 5:49–68, 1989.

[Don92] D. L. Donoho. Interpolating wavelet transforms. Technical report, Technical Report 408, Department of Statistics, Stanford University, 1992.

[Mal97] S. G. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 1997.